



LAUREA

AMMATTIKORKEAKOULU

Yhdessä enemmän

Palvelinvalvonnan kehittäminen yrityksessä RELEX Oy

Mattila, Sami

2015 Leppävaara

Laurea-ammattikorkeakoulu
Laurea Leppävaara

Palvelinvalvonnan kehittäminen yrityksessä RELEX Oy

Sami Mattila
Tietojärjestelmäosaaminen YAMK
Opinnäytetyö
Huhtikuu, 2015

Sami Mattila

Palvelinvalvonnan kehittäminen yrityksessä RELEX Oy

Vuosi	2015	Sivumäärä	56
-------	------	-----------	----

Opinnäytetyöni on laadullinen tutkielma, jonka tavoitteena on kehittää palvelinvalvontaa yrityksessä RELEX Oy. Opinnäytetyötä tehdään, koska yritys on kokenut, että nykyinen palvelinvalvonta ei tule riittämään yrityksen tarpeisiin tulevaisuudessa ominaisuuksiensa sekä skaalautuvuutensa puolesta. Yrityksen nykyisen palvelinvalvonnan suurimpiin ongelmiin kuuluu, että se koostuu kahdesta eri tarkoituksia palvelevasta palvelinvalvontaohjelmistosta, jonka ylläpito- ja kehitystyö ovat haasteellisia. Lisäksi taltioitun datan esitysmuoto ei ole tarpeeksi monipuolista. Nykyisen palvelinvalvontajärjestelmän eduksi on koettu toimintavarmuus sekä kerätyn tiedon luotettavuus.

Opinnäytetyöni on tehty mukaillen usean tutkimusmenetelmän kehikkoa tietojärjestelmätutkimuksessa. Kehikon ideana on yhdistää vuorovaikutuksen avulla kehikon vaiheet, jotka ovat teoriataustan muodostus, testaus, tutkimus sekä järjestelmäkehitys. Opinnäytetyöni tulokset antavat järjestelmäkehitysvaiheeseen palvelinvalvonnan kehitysehdotuksen. Opinnäytetyöni teoriatausta kertoo järjestelmänvalvonnasta, valvontasovelluksista ja ennustamisesta järjestelmänvalvonnassa.

Opinnäytetyössäni esitetään vastaukset tutkimuskysymyksiin miksi palvelinvalvonta on tärkeä osa järjestelmänvalvontaa ja liiketoimintaa sekä miten palvelinvalvontaa voidaan kehittää. Opinnäytetyöhön tehdyissä kahdessa tapaustutkimuksessa on käytetty tutkimusmenetelminä haastattelua ja havainnointia. Ensimmäinen tapaustutkimus tutkii palvelinvalvonnan nykytilaa yrityksessä ja toinen tapaustutkimus tutkii miten ennalta valitut valvontasovellukset sopivat yrityksen tarpeisiin vaatimusmäärittelyn avulla. Tämän lisäksi sovellustestaus ja teorian tieto ovat antaneet arvokasta lisätietoa usean tutkimusmenetelmän kehikon periaatteen mukaisesti. Opinnäytetyössä kerätty data on analysoitu käyttäen vuorovaikutteisen analysointiotteen menetelmää.

Opinnäytetyön tuloksena on käyttöönottoehdotus Zabbix palvelinvalvontasovelluksesta, joka täyttää yrityksen vaatimukset palvelinvalvonnalle. Tämän lisäksi selkiytetään minkälainen merkitys palvelinvalvonnalla on RELEX Oy:lle järjestelmänvalvonnan ja liiketoiminnan näkökulmista. Opinnäytetyön tulosten avulla yrityksen palvelinvalvontatoimintaa sekä liiketoimintaa voidaan tehostaa merkittävästi käyttämällä henkilö- ja laitteistoresursseja oikein, ehkäisemällä ongelmatilanteiden muodostumista, sekä tehostamalla ja monipuolistamalla selvitys ja valvontatoimintaa.

Asiasanat: järjestelmänvalvonta, palvelinvalvonta, valvontasovellus, valvontajärjestelmä

Sami Mattila

Developing server monitoring in RELEX Oy

Year	2015	Pages	56
------	------	-------	----

The thesis is a qualitative research with the objective of developing system monitoring in RELEX Ltd. The thesis was conducted due to the need of developing the current monitoring system to meet the ever-increasing future needs such as features and scalability of the monitoring domain. The most notable problem of the current monitoring system in use is that it consists of two different monitoring applications, which both require a lot of administration and development work. In addition, the presentation of the collected data is not versatile enough. The current monitoring system has been praised for data and application reliability.

The thesis has adapted the multimethodological approach in information systems research. The general idea of the approach is to combine the phases of the research using interaction. The interacting phases are theory building, experimentation, observation and systems development. The results of the thesis are proposals for developing the system monitoring in the systems development phase. The theory of the thesis consists of system administration, system monitoring and forecasting in system monitoring.

The thesis answers the research questions why is system monitoring an important part of system administration and business and also how can the system monitoring be developed. The thesis includes two case studies using interview and observation as research methods. The first case study is about understanding the current state of system monitoring in RELEX. The second case study is about finding a suitable system monitoring solution using software requirements specification. In addition, experimentation and theory building have played an important role in developing valuable information while applying the interactive multimethodological approach. The data gathered in the thesis was analyzed using an interactive analysis method.

The results of the thesis are an introduction proposal of Zabbix system monitoring framework, which fulfills the RELEX Ltd's monitoring requirements and also a clarification of what is the significance of system monitoring for RELEX Ltd. in system administrating and business perspective. The results of the thesis will significantly enhance system monitoring and business by optimizing human and system resource requirements, preventing problem situations, enhancing and diversifying problem solving and monitoring.

Keywords: system monitoring, server monitoring, monitoring application, monitoring system

Sisällys

1	Johdanto.....	6
2	Teoreettinen viitekehys	7
2.1	Järjestelmänvalvonta.....	7
2.2	Valvontasovellus.....	9
2.2.1	Tilannetietojen kerääminen ja tallentaminen	10
2.2.2	Valvottavat asiat.....	13
2.3	Ennustaminen järjestelmänvalvonnassa	15
3	Palvelinvalvonnan nykytilanne yrityksessä	17
4	Tutkimuksen kuvaus	19
4.1	Tutkimuksen teema ja tutkimuskysymykset	19
4.2	Käytettyjen tutkimusmenetelmien kuvaus	20
4.2.1	Tapaustutkimuksen vaiheet	21
4.2.2	Haastattelu	23
4.2.3	Havainnointi	25
4.3	Datan kuvaus ja datan kerääminen.....	26
4.4	Analysointi	27
4.5	Tutkimuksen kulku	33
4.5.1	Tapaustutkimus 1	38
4.5.2	Tapaustutkimus 2	41
5	Tulokset.....	48
6	Keskustelu	53
6.1	Validiteetti ja reliabiliteetti	53
6.2	Tulosten pohdinta ja hyödynnettävyys	54
6.3	Jatkotutkimusehdotukset	56
	Lähteet	57
	Kuvat.....	59
	Taulukot	59
	Liitteet.....	59

1 Johdanto

Lähes kaikki tuotantokäytössä olevat tietojärjestelmät ovat jonkinlaisella palvelimella, jota järjestelmänvalvojat valvovat joko manuaalisesti tai automaattisen palvelinvalvonnan avulla. Palvelinvalvontaa suoritetaan, jotta voidaan olla varmoja että tarjottavat tietojärjestelmät toimivat asianmukaisesti. Tehokas palvelinvalvonta ennakoi ja estää ongelmien syntymisen tai vähintäänkin auttaa ongelmien määrittämisessä ja ratkaisemisessa. Palvelinvalvonta voi olla tehokasta jos se on toteutettu oikein.

Tämän opinnäytetyön tarkoituksena on toimeksiantona kehittää RELEX Oy:n palvelinvalvontaa siten, että se vastaa yrityksen tarpeita ja on tehokasta. Opinnäytetyö auttaa ymmärtämään palvelinvalvonnan merkityksen niin järjestelmänvalvonnan kuin liiketoiminnan osalta. Opinnäytetyötä tehdään, koska yrityksessä on koettu, että nykyinen palvelinvalvontajärjestelmä ei riitä nykyisiin ja tuleviin kasvaviin tarpeisiin tulevaisuudessa huonon skaalautuvuuden sekä ominaisuuksiensa puolesta. Nykyinen palvelinvalvontajärjestelmä on yrityksen itse tuottama, jonka kehitys on aikaa sekä resursseja vievää. Näistä syistä on koettu tarpeelliseksi tutkia, mitkä ovat yrityksen vaatimukset palvelinvalvonnalle. Vaatimusmäärittäminen avulla pystytään tutkimaan täyttääkö jokin valmis palvelinvalvontaratkaisu asetetut vaatimukset vai jatketaan nykyisen palvelinvalvontajärjestelmän kehitystä.

Kiinnostavaa palvelinvalvonnasta ja palvelinvalvontasovelluksista tekee niiden monimuotoisuus. Palvelinvalvontasovellukset vastaavat erilaisiin tarpeisiin ja antavat samoihin ongelmiin erilaisia ratkaisuja. Palvelinvalvontasovelluksien on oltava riittävän monipuolisia, jotta ne pystyvät tukemaan useaa eri valvottavaa palvelinjärjestelmää sekä samalla muuntautumiskykyisiä, jotta pystytään täyttämään yksilöityjä tarpeita. Valitsin opinnäytetyöni aiheen, koska palvelinvalvonnan kysyntä kasvaa kaikkialla jatkuvasti ja viime aikoina ilmaiset palvelinvalvontasovellukset ovat kehittyneet paljon, mikä tekee niiden vertailun mahdolliseksi sekä järkeväksi. Opinnäytetyön aihe myös kiinnostaa ja kehittää minua ammatillisesti sekä palvelee nykyistä työnkuvaani tietojärjestelmien kehittäjänä.

Opinnäytetyö on kvalitatiivinen tutkimus, jonka tavoitteena on kehittää yrityksen palvelinvalvontaa vastaamalla opinnäytetyön tutkimuskysymyksiin. Opinnäytetyö on tehty mukaillen usean tutkimusmenetelmän kehikkoa tietojärjestelmätutkimuksessa, joka on esitetty Nunnallyn, Chinin ja Purdinin (1991) toimesta artikkelissa Systems Development in Information Systems Research. Kehikon ideana on yhdistää vuorovaikutuksen avulla kehikon vaiheet, jotka ovat teoriataustan muodostus, testaus, tutkimus sekä järjestelmäkehitys. Opinnäytetyössä ei toteuteta järjestelmäkehitysvaihetta, vaan se on rajattu pois. Opinnäytetyön tulokset antavat ehdotuksen palvelinvalvonnan kehitykseen. Opinnäytetyön teoriatausta auttaa ymmärtämään järjestelmänvalvontaa sekä palvelinvalvontasovelluksia. Opinnäytetyössä etsitään vastausta

tutkimuskysymyksiin miksi palvelinvalvonta on tärkeä osa järjestelmänvalvontaa ja liiketoimintaa sekä miten palvelinvalvontaa voidaan kehittää. Opinnäytetyöhön tehdyissä kahdessa tapaustutkimuksessa on käytetty tutkimusmenetelminä haastattelua sekä havainnointia. Tämän lisäksi sovellustestaus sekä teorian tiedon läpikäynti on antanut arvokasta lisätietoa usean tutkimusmenetelmän kehikon periaatteen mukaisesti.

Opinnäytetyössä kerätty kvalitatiivinen data on analysoitu vuorovaikutteisen analysointiotteen menetelmällä, joka on esitetty kirjassa *Qualitative Data Analysis* (Miles, Huberman & Saldaña 2014). Opinnäytetyön tuloksena annetaan suositus käyttöönotosta yrityksen vaatimukset toteuttavaan palvelinvalvontasovellukseen. Tämän lisäksi selkiytetään yritykselle palvelinvalvonnan merkitystä järjestelmänvalvonnassa sekä liiketoiminnassa. Tuloksilla on merkittävä vaikutus, sillä niiden avulla voidaan tehostaa palvelinvalvontaa ja siten vaikuttaa positiivisesti yrityksen liiketoiminnalliseen menestykseen. Lopuksi esitetään jatkotutkimusehdotukset, joiden avulla voidaan kehittää uutta palvelinvalvontajärjestelmää vastaamaan vieläkin paremmin myös tulevia vaatimuksia.

2 Teoreettinen viitekehys

Opinnäytetyön aihe vaatii ymmärrystä IT-alasta ja tekniikasta, sillä sen luonne on tutkia palvelinvalvontaa, joka on varsin tekninen aihe. Teoreettinen viitekehys -luku antaa tarvittavan lisäymmärryksen opinnäytetyössä esiintyvistä aihealueista sekä alasta, jota opinnäytetyössä tutkitaan. Teoreettinen viitekehys käsittelee mitä on järjestelmänvalvonta ja miten se on syntynyt. Tämän lisäksi kerrotaan mikä on valvontasovellus, miten se kerää tilannetietoja ja mitä asioita tulisi valvoa. Lopuksi käydään läpi ennustamisen merkitys ja toteutus järjestelmänvalvonnassa.

2.1 Järjestelmänvalvonta

Ennen UNIX-käyttöjärjestelmän yleistymistä tietokoneet olivat tarkoitettu vain yhtä tarkoitusta varten eikä niitä voinut useampi ihminen käyttää samaan aikaan. Silloin tietokoneen käyttäjiä kutsuttiin vielä operaattoreiksi järjestelmänvalvojien sijaan. UNIX syntyi vuosien 1969 ja 1973 välisenä aikana, mutta vasta 1980-luvun alussa UNIXin käyttö levisi yliopistoihin kun opiskelijat kiinnostuivat uudesta ja ihmeellisestä teknologiasta. Usean käyttäjän moniajojärjestelmät vaativat tuolloin jatkuvaa järjestelmänvalvontaa, sillä vikatilanteet eivät olleet harvinaisia sovellus- tai laitetasolla. Tällöin yliopistot jakoivat toistensa kanssa järjestäytyneesti prosesseja, standardeja ja parhaita käytäntöjä järjestelmien ylläpitoon. Samaa aikaa yliopistoissa ensimmäiset opiskelijat alkoivat työskennellä uudella työnimikkeellä, joka oli järjestelmänvalvoja. Tästä katsotaan kyseisen ammattinimikkeen saaneen alkunsa. Järjestelmänvalvonta on toimintaa, jossa kerätään järjestelmästä systemaattisesti tietoa määritel-

lyistä asioista. Järjestelmänvalvoja voi tarkastella tätä tietoa ja ilmoittaa sidosryhmille tilanteen tai puuttua järjestelmän toimintaan jos tilanne sitä vaatii. (Nemeth, Snyder, Hein & Whaley 2011, 1264 - 1269; Powers, Peek, O'Reilly & Loukides 2003, 3 - 5.)

Tuotantokäytössä olevat järjestelmät vaativat usein joko automaattista tai käsin tehtävää valvontaa. Valvontaa tehdään, jotta järjestelmissä ilmenevät häiriötilanteet tai niiden alut voidaan ratkaista. Järjestelmää käyttävät ihmiset eivät yleensä ilmoita ongelmista ellei kyse ole vakavasta käyttöä haittaavasta ongelmasta. Monesti ongelman kiertäminen on helpompaa kuin ongelmasta raportointi, minkä vuoksi käyttäjät usein valitsevat heille helpomman tavan jatkaakseen työskentelyä. (Nemeth ym. 2011, 5.)

Koska usein ongelmat ovat kierrettävissä, järjestelmänvalvojat eivät välttämättä koskaan saa tietoa, että järjestelmässä on ollut jokin vialla. Kun pienikin ja helposti korjattava ongelma ei ole tiedossa, järjestelmänvalvojat eivät voi tehdä asialle mitään. Valvonta on tärkeä ja perustavanlaatuinen osa yrityksen tarjoamaa IT-palvelua. Sanotaankin, että ”jos et valvo sitä, et voi hallinnoida sitä”. IT-alalla on yleistä, että jos sähköisessä palvelussa on katkos, se johtaa helposti taloudellisten ansioiden menetykseen. Valvontasovelluksia voidaan hyödyntää automaattisesti havainnoimalla ja korjaamalla ongelmia, tunnistamalla ongelman lähteitä ja enustamalla sekä välttämällä tulevia ongelmia. (Limoncelli, Hogan & Chalup 2013, 523.)

Järjestelmänvalvontaa on kolmea erilaista. Näitä ovat historiallinen, reaaliaikainen sekä enustava valvonta. Kaikilla valvonnan muodoilla on oma tarkoituksensa ja ne vastaavat eri tarpeisiin. (Limoncelli ym. 2013, 523.) Järjestelmää tulisi säännöllisesti valvoa varmistaen, että sen tarjoamat palvelut toimivat oikein ja ovat käytettävissä. Useimmiten varsinkin loki-tiedostoja tulisi valvoa. Loki-tiedostot kertovat järjestelmän tilasta sekä tapahtumista. Valvonnan avulla ongelmat tai niiden alut voidaan havaita mahdollisimman aikaisin. (Nemeth ym. 2011, 5.)

Historiallisessa valvonnassa tarkastellaan taltioitua dataa. Taltioitu data voi olla esimerkiksi järjestelmän saatavuus- tai suoritusaste jollakin ajanhetkellä. Dataa tarkastelemalla voidaan vetää johtopäätöksiä kuten, että palvelu oli käytettävissä 99,9% ajasta edellisen kuukauden aikana. Suoritusastetta voidaan tarkastella suuremmalta aikaväliltä ja vetää esimerkiksi johtopäätöksiä siitä milloin järjestelmän suorituskyky on alkanut kärsiä alitehoisen laitteiston vuoksi.

Reaaliaikainen valvonta hälyttää järjestelmänvalvojia virhetilanteista heti, kun tieto siitä saadaan. Reaaliaikaisessa valvonnassa on kaksi eri komponenttia. Ensimmäinen komponentti havaitsee ongelmat niiden tapahtuessa ja toinen komponentti hälyttää asianomaisia, kun ensimmäinen komponentti havaitsee ongelman. (Limoncelli ym. 2013, 524.)

Näiden kahden pääasiallisten valvontamallien lisäksi alkaa resurssien ennustamisen tarve näkyä järjestelmänvalvonnassa. Kerätyn historiadatan avulla on mahdollista muodostaa aikasarjaennustemalleilla ennusteita järjestelmän resurssikehityksestä ja siten saada mahdollisia vikatilanteita kiinni ennen kuin niistä on muodostumassa ongelmia. Tämän lisäksi ennusteiden avulla voidaan tehdä valvottavan järjestelmän resurssisuunnittelua. (Makridakis, Wheelwright & Hyndman 1998, 16.)

2.2 Valvontasovellus

Valvontasovellus on kohdeympäristössä suoritettava sovellus, joka valvoo ennalta määritettyjä asioita. Valvontasovellusta voidaan käyttää ongelmien havaitsemiseksi sekä korjaamiseksi, ongelmien aiheuttajien löytämiseksi, ongelmien välttämiseksi ja ennustamiseksi sekä järjestelmästä dataa kerääväksi järjestelmänvalvojan työkaluksi. Valvontasovellus on tärkeä osa järjestelmänvalvontaa sekä tarjottavaa ammattimaista järjestelmävalvontapalvelua. Valvontasovelluksen avulla tähdätään, että valvottava järjestelmä olisi aina käyttäjien tavoitettavissa. Valvontasovellus pitää huolta järjestelmästä ja täten auttaa yritystä pitämään huolta siitä, että liikevaihto ei kärsi katkoksien vuoksi. (Limoncelli ym. 2013, 523 - 535.)

Palvelinvalvontaa voidaan tehdä kolmella eri kypsyystasolla. Palvelinvalvonnan kypsyystaso riippuu siitä millä kehitysasteella palvelinvalvonta on. Kypsyystasoja ovat passiivinen, aktiivinen sekä proaktiivinen. Passiivisessa palvelinvalvonnassa tapahtumat kirjataan valvontasovellukseen ja järjestelmänvalvoja suorittaa valvonnan myöhemmin käyden läpi kirjatut tapahtumat. Aktiivinen palvelinvalvonta havainnoi järjestelmän tapahtumia ja ilmoittaa järjestelmänvalvojille virhetilanteista reaaliajassa. Proaktiivinen palvelinvalvonta toimii kuten aktiivinenkin, mutta lisäksi yrittää korjata virhetilanteita esiasetettujen määritysten mukaisesti. (Chuvakin 2010, 14 - 15.) Tämän lisäksi on kolme erilaista valvontasovelluksen varoitusta, jotka ovat reaktiivinen, proaktiivinen sekä ennustava. Reaktiivinen varoitus tarkoittaa sitä, että järjestelmänvalvojaa varoitetaan, kun virhe on tapahtunut, esimerkiksi kun järjestelmä on kaatunut. Proaktiivinen varoitus lähtee järjestelmänvalvojille ennen kuin järjestelmä kaatuu, esimerkiksi tapauksissa joissa palvelimen vapaata kiintolevytilaa on jäljellä 5%. Ennustava varoitus pyrkii tutkimaan miten mitattava asia on käyttäytynyt menneisyydessä ja riittävätkö nykyiset resurssit järjestelmän toimimiseen jollakin tulevilla aikavälillä, esimerkiksi onko vapaata kiintolevytilaa tarpeeksi kuukauden päästä. (Jong 2009, 112.)

Valvontasovelluksen on oltava hyvin joustava, koska valvontasovelluksen tarkoituksena on varmistaa ja raportoida muiden järjestelmien toimivuutta. Valvontasovelluksen on ymmärrettävä kaikkia valvontaan sopivia protokollia, joita valvottavat järjestelmät käyttävät. Ollakseen monipuolisia, monet valvontasovellukset yrittävät tietää etukäteen, mitä järjestelmän-

valvojat haluavat valvoa ja liittää nämä ominaisuudet valvontasovelluksen vakioasennukseen. Kaiken kattavat valvontasovellukset, jotka tietävät miten valvoa kaikkea yritykselle tärkeitä, pitää voida myös kehittää järjestelmänvalvojien toimesta. Kehitystä tarvitaan, kun jokin uusi valvottava asia ilmenee, jota aiemmin ei ole otettu huomioon. Ilman liitännäisiä valvontasovellus harvoin sopii sellaisenaan mihinkään organisaatioon. Valvontasovelluksiin voi valvontajärjestelmästä riippuen luoda yleensä vapaasti valittavalla ohjelmointikielellä lisää valvontalogiikkaa. Valvontalogiikkaa lisätään noudattaen protokollaa, jolla keskustella valvontasovelluksen kanssa. (Josephsen 2007, 13 - 14.)

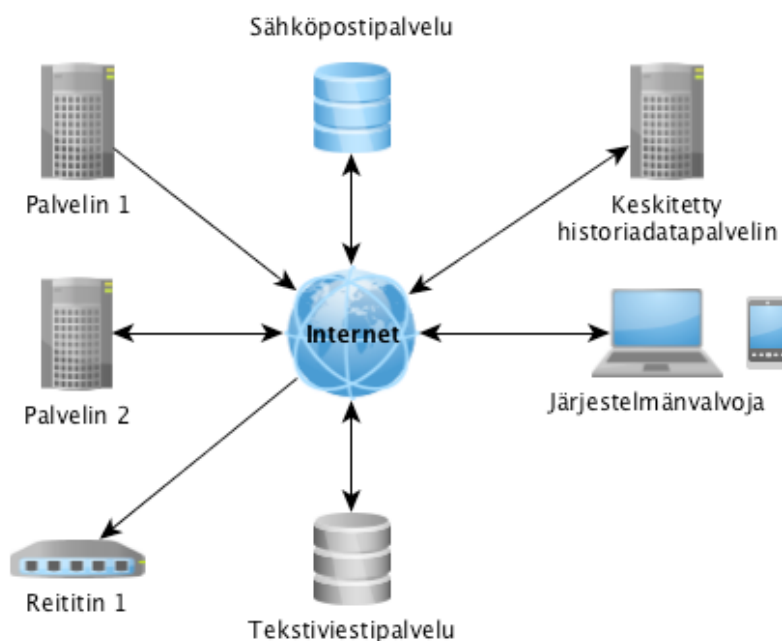
Avoimen lähdekoodin valvontasovelluksen hyötyihin voidaan lukea, että käyttäjät voivat tutustua ohjelmakoodiin ja sitä myöten henkilöt tai yhteisöt voivat kehittää lähdekoodikantaa omiin tarpeisiinsa. Tällöin koko yhteisö hyötyy parannetusta lopputuotteesta, lähdekoodista sekä jaetusta informaatiosta. Avoimen lähdekoodin peruseräitä ovat myös, että avoimen lähdekoodin sovellukset ovat ilmaisia. Avoimen lähdekoodin etuja suljettuun lähdekoodiin verrattuna on, että projektit eivät kuole jos niiden alkuperäiset kehittäjät estyvät tavalla tai toisella kehittämästä lähdekoodikantaa. Jotkut voivat kysyä, että jos sovellus on ilmainen, mistä saa tukea ongelmatilanteissa. Avoimen lähdekoodin useimmat lisenssit antavat yrityksille mahdollisuuden myydä avoimen lähdekoodin sovelluksiin tukea, tähän jopa kannustetaan, sillä se vain lisää avoimen lähdekoodin suosiota ja sovellusten käytettävyyttä. Sovelluksesta tai koodista itsestään ei siis tulisi maksaa kenellekään. (Farkas 2011, 36.)

Valvontatoiminnassa on tärkeää olla jokin tapa valvoa, että valvontajärjestelmä tekee työtään. Valvontajärjestelmän kaatuminen voi aiheuttaa monenlaista harmia, kuten menetetyt hälytykset, jotka johtavat valvottavien järjestelmien epävakauksiin sekä esimerkiksi palvelutasoraportoinnin keskeytyminen. Palvelutasosopimus on palveluntarjoajan sekä asiakkaan välinen sopimus, jossa määritetään vaatimustasot tarjottavalle palvelulle. Jos vaatimustaso alitetaan, on tästä palvelutasosopimuksessa sovittu sanktio. Valvontajärjestelmien vastuu on mitata ja raportoida tätä palvelutasoa. Jos valvontajärjestelmä ei toimi, ei dataa palvelutason mittaamiseen kerätä. On siis erityisen tärkeä luoda jokin tapa varmistaa valvontasovelluksen toiminta, sekä varmistaa valvonnan jatkuvuus esimerkiksi varavaltajärjestelmän avulla. (Josephsen 2007, 11 - 12.)

2.2.1 Tilannetietojen kerääminen ja tallentaminen

Valvontasovellukset keräävät tilannetietoa keskitettyyn valvontasovellusjärjestelmään kolmella eri tavalla. Ensimmäisessä tavassa (pull) keskitetty valvontasovellusjärjestelmä hakee tilannetiedon valvottavilta kohteilta. Tilannetiedon haku tehdään protokollalla, jota kumpikin järjestelmä ymmärtää. Kuvassa 1 tätä on havainnointu siten, että keskitetty historiadatapalvelin hakee tapahtumadatan reitittimeltä 1 SNMP-protokollan (Simple Network Management

Protocol) avulla. Toisessa tavassa (push) valvottavat kohteet ilmoittavat tilannetietonsa periodisesti keskitetylle historiadatapalvelimelle. Tilannetiedon ilmoitus tapahtuu valvottavaan järjestelmään asennettavan valvontasovellus-agentin avulla. Kuvassa 1 tätä on havainnoitu siten, että palvelin 1:n valvontasovellus-agentti lähettää tilannetietonsa keskitetylle historiadatapalvelimelle. Kolmas tapa on näiden yhdistelmä, keskitetty historiadatapalvelin voi kysyä tilannetietoja ja palvelimen valvontasovellus-agentti lähettää periodista tilannetietoa keskitetylle historiadatapalvelimelle. (Martin-Flatin 1998, 1 - 2, 7.)



Kuva 1: Valvontasovelluksen tilannetietojen keräys.

Tilannetietojen keräystapa riippuu valvottavasta järjestelmästä ja infrastruktuurista johon se on kytketty. Verkon aktiivilaitteet saattavat ymmärtää vain SNMP-protokollaa, joka on verkkojen hallinnassa käytettävä tietoliikenneprotokolla. Verkkolaitteisiin ei voi asentaa valvontasovellus-agenttia, joten ainoaksi vaihtoehdoksi jää ensimmäinen tapa. Verkon infrastruktuuri saattaa evätä valvottavalle palvelimelle tulevat tai sieltä lähtevät yhteydet, jolloin täytyy tapauskohtaisesti miettiä haetaanko vai lähettääkö valvottava järjestelmä tilannetietoa valvontasovellus-agentin avulla. (Martin-Flatin 1998, 5.)

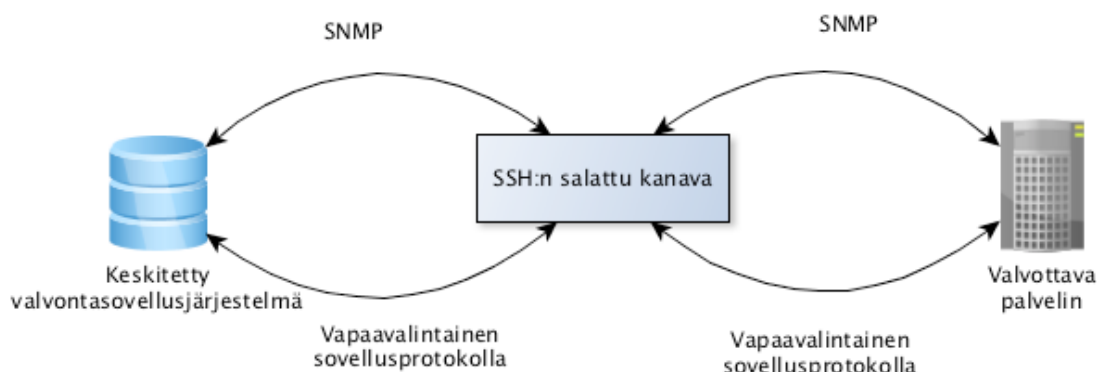
Keskitetyn valvontasovellusjärjestelmän hakiessa tilannetiedon valvottavalta kohteelta, tehtävään vaadittava prosessointiteho ja tietojen haku rasittaa keskitetyn valvontasovellusjärjestelmän palvelinta enemmän kuin valvottavaa kohdetta. Keskitetty valvontasovellusjärjestelmä noutaa periodisesti esiasetettuja tietoja valvottavalta järjestelmästä esimerkiksi SNMP tai SSH protokollan ylitse. Tämän keräystavan eduksi lasketaan valvottavan kohteen vähäinen kuormitus ja huonoksi puoleksi, että jos kerättävää dataa on paljon, keskitetty valvontajär-

jestelmä ei pahimmassa tapauksessa ehdi hakea dataa sille annetussa aikamääreessä. (Martin-Flatin 1998, 7 - 8.)

Roolit muuttuvat päinvastaiseksi, kun agentti lähettää tilannetiedon valvottavalta kohteelta keskitettyyn valvontajärjestelmään. Tätä tapaa käytetään usein kun palomuuuri estää Internetistä palvelimelle tulevat pyynnöt, mutta antaa palvelimen kommunikoida Internetissä tai muualla verkossa sijaitsevien muiden palvelimien kanssa. Roolien muuttuessa täytyy agentin olla aktiivinen osapuoli. Aktiivisen osapuolen tulee tietää mitä tilannetietoja lähetetään keskitettyyn valvontajärjestelmään ja kuinka usein. (Martin-Flatin 1998, 3, 7 - 8.)

SNMP-protokollan avulla voidaan kerätä tietoa laitteen toiminnasta, jossa on hallinta-agentti (SNMP-agent) käytössä. Kerättyjä tietoja kysellään hallinta-ohjelmalla (management station). Hallinta-ohjelman ja hallinta-agentin välisessä kommunikaatiossa käytetään SNMP-protokollaa. SNMP on sovellustason protokolla, joka määrittää missä muodossa hallinta-agentti suorittaa kyselyn MIB-tietokantaan (Management Information Base), johon on hallinta-agentin määrittelyn mukaisesti kerätty hallinta-ohjelmalla kyseltävä data. Hallinta-ohjelman suorittaman datankeräyksen lisäksi hallinta-agentti voidaan määrittää lähettämään tapahtumia (event), joilla on jokin kynnyсарvo (threshold), hallinta-asemalle trap-sanomina. Trap-sanomia käytettäessä, on vastaanottavassa palvelinvalvontasovelluksessa oltava tuki SNMP-protokollan trap-sanoman vastaanottoon. Useimmat verkon aktiivilaitteet, kuten kytkimet, tukevat SNMP-protokollaa. Verkkolaitteista usein mitataan pullonkauloja laitteen läpi menevässä tietoliikenteessä tai aktiivilaitteen kykyä suoriutua sen kautta kulkevasta tietoliikenteestä. Monet palvelinvalvontasovellukset sisältävät valmiita verkon aktiivilaitteille suunnattuja testejä, jotka mittaavat laaja-alaisesti laitteen ja tietoliikenteen tilaa. (Simoneau 1999, 34 - 40.)

SSH eli Secure Shell on sovellustason protokolla, jonka avulla liikennöinti salataan automaattisesti niin, että käyttäjä ei sitä käytännössä edes huomaa. SSH ei ole komentotulkki, kuten usein luullaan, vaan SSH toimii palvelinohjelmisto asiakasohjelmisto periaatteella. SSH-protokolla käyttää moderneja, turvallisia kryptaus algoritmeja, ja se on käytössä useimmissa suuryrityksissä sekä tuotantoympäristöissä. SSH-protokollan tehtävänä on muodostaa salattu kanava asiakkaan ja palvelimen välille. SSH-protokolla toteuttaa turvallisen todennuksen, kryptauksen sekä lähetetyn tai vastaanotetun datan eheyden eli muuntamattomuuden. SSH:n tehtävänä on varmistaa, että kummatkin yhteyden osapuolet ovat keitä he väittävät olevansa ja että välitetty data on eheää ja sitä ei ole pystytty salakuuntelemaan. (Barret & Silverman 2001. 1 - 5.) Kuvassa 2 on havainnollistettu keskitetyn valvontasovellusjärjestelmän ja valvottavan palvelimen välistä tietoliikennettä SSH:n salattua kanavaa hyödyntäen.



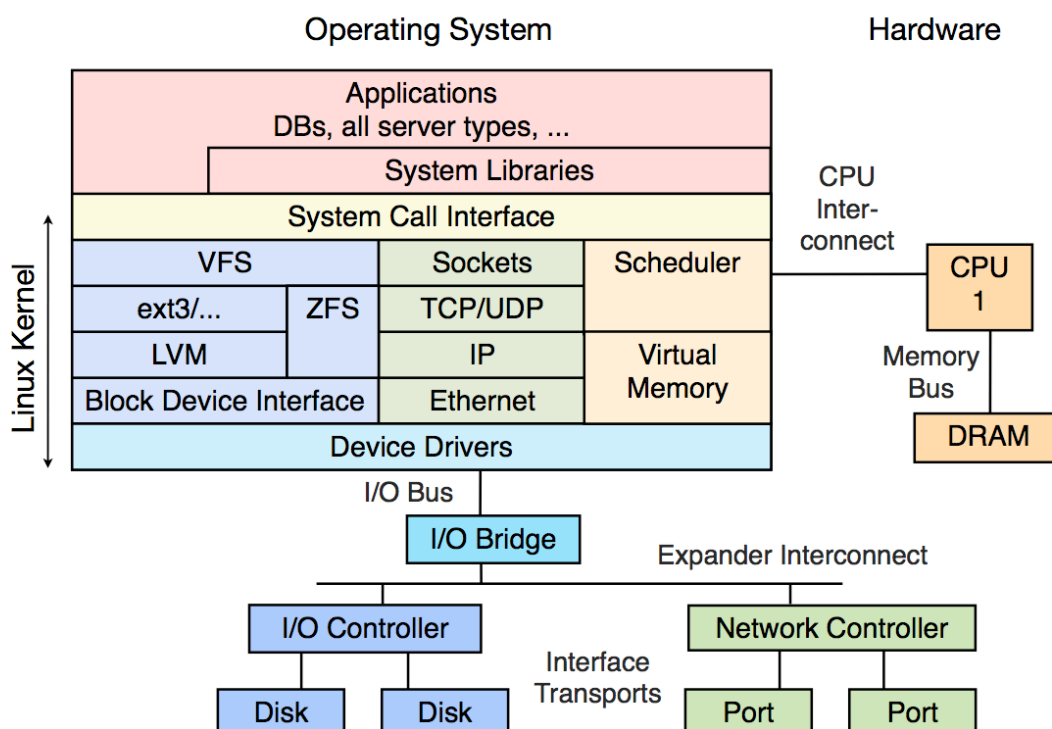
Kuva 2: SSH ja salattu kanava.

Tilannetiedot tallennetaan tietokantaan myöhempää tarkastelua varten. Tallennetuista tilannetiedoista voidaan johtaa analyysiä tai visualisoida sitä. Koska mitattavia tilannetietoja sekä valvottavia palvelimia on paljon ja tilannetietojen keräys voi olla tiheää, tallennetaan tietokantaan kokonaismäärältään tilannetietoa hyvin paljon. Koska lähihistorian tilannetiedot ovat merkittävämpiä, on yleistä että mitä vanhempaa dataa tarkastellaan, sitä epätarkempaa se on. Kun tilannetieto vanhenee, dataa harvennetaan ottamalla siitä mediaani, keskiarvo, minimi tai maksimi joltain aikaväliltä ja korvaamalla aikavälin tallennetut tilannetiedot saaduilla arvoilla. (Jugel, Jerzak & Markl 2014, 797 - 800, 807.) Tehty datan redusointi parantaa keskitetyn historiadatapalvelimen suorituskykyä sekä vähentää kiintolevytilan käyttöastetta.

2.2.2 Valvottavat asiat

Vaikka valvottavien asioiden merkitys on organisaatiokohtaista, valvontaa suoritetaan yleensä kolmesta eri syystä. Järjestelmänvalvojat käyttävät valvontajärjestelmiä valvoakseen järjestelmän tilaa, mitatakseen suorituskykymuutoksia sekä vianetsinnän avuksi. Kapasiteettisuunnittelua tekevät järjestelmänvalvojat käyttävät valvontajärjestelmiä mitatakseen ja ennustakseen järjestelmien kapasiteetin riittävyyttä. (Andersen 2004, 1.)

Monet asiat voivat aiheuttaa palvelimissa hitautta. Tällaisia asioita ovat esimerkiksi käyttöjärjestelmä, laitteet, sovellukset tai näiden konfiguraatio. Edellä mainittujen asioiden moitteeton toiminta on toisistaan riippuvaisia. Tästä syystä on tärkeä mitata kaikkea, jotta tiedetään missä vika oikeasti piilee. (Gregg 2014, 4 - 5.) Kuvassa Kuva 3: Esimerkki palvelimien seurattavista komponenteista on havainnollistettu tyypillistä palvelinta Linux-käyttöjärjestelmällä. Kuvassa laitteita sekä protokollia on lokeroitu loogisiin osiin ja se auttaa havainnollistamaan laitteiden ja protokollien välisiä riippuvuussuhteita.



Kuva 3: Esimerkki palvelimien seurattavista komponenteista (Gregg 2014).

Lokitiedostojen säännöllisen seurannan avulla voi ilmetä tietoturvahyökkäyksiä tai niiden yrityksiä sekä järjestelmän vakaata toimintaa haittaavia tapahtumia. Lokitietoja voidaan käyttää apuna vianselvityksessä. Valvonnan pitäisi olla mahdollisimman yksityiskohtaista, jotta vikatilanteen syy pystytään rajaamaan mahdollisimman tarkasti. Tarkasti rajattuihin vikojen syihin on helpompi hakea ratkaisua. Esimerkiksi prosessorin kuormaa tulisi mitata säie-tasolla prosessorin kokonaiskuorman sijaan, jotta voidaan havaita säikeisiin liittyvät skaalautuvuusongelmat. (Gregg 2014, 220.)

Siltä osin kuin on mahdollista, palvelinympäristöjen määritysten olisi hyvä olla mahdollisimman yhdenmukaisia, jotta virhetilanteet minimoitaisiin ja järjestelmien käyttäytymistä voidaan ennakoida. Valvontasovelluksen on hyvä seurata määritysten tilaa sekä ohjelmistoversioita ja raportoida, jos nämä poikkeavat normaalista. (Gregg 2014, 220.)

Valvontatarpeita on myös tärkeä katsoa käyttäjien ja liiketoiminnan näkökulmasta. Järjestelmänvalvojille on tärkeää valvoa laitteen kaikkia mittareita, mutta käyttäjille ja liiketoiminnalle tärkeitä on nähdä tietoja, kuten kuinka nopeasti järjestelmä vastaa asiakkaalle. (Gregg 2014, 67.)

2.3 Ennustaminen järjestelmänvalvonnassa

Laskettujen ennusteiden avulla järjestelmänvalvovat voivat huomata tulevat ongelmat jo etukäteen ja reagoida niihin, jotta ongelmat lieventyisivät tai eivät toteutuisi. Ennalta tietämättömän tapahtuman toteutumisen ja tapahtumaan reagoinnin välissä on aikaväli, joka on pääasiallinen syy ennustamiseen ja suunnittelemiseen. Jos kyseinen aikaväli on olematon, ei ole syytä ennustaa tapahtumia. Aikavälin ollessa pitkä ja tapahtuman seuraukset tiedossa, korostuu ennustamisen tarve. (Makridakis ym. 1998, 2, 16.)

Ennustamisessa on hyvä muistaa, että on valvottavissa olevia asioita, sekä ennakoimattomia asioita. Ennakoimattomat asiat johtuvat usein ulkoisista syistä joita ei voi valvoa tai vaikuttaa. Hyvä järjestelmän toiminta riippuu kummankin tyyppisistä asioista. Valvottavissa olevia asioita tulee ennustaa sekä ennakoimattomien asioiden tapahtuessa tehdä päätöksiä. Hyvä suunnittelu yhdistää nämä kaksi asiaa toisiinsa. (Makridakis ym. 1998, 2.)

Ennustamista voi tehdä monin eri tavoin. Tapojen monimutkaisuus vaihtelee helposta edellisen havainto ennusteksi -mallista monimutkaisiin neuroverkkoihin. On tärkeää valita omaan dataan ja käyttötarpeisiin sopiva ennustemalli, jolloin saadaan hyviä ennusteita. Määrämuotoisen, kvantitatiivisen datan ennustamisen lisäksi tehdään usein myös kvalitatiivista, laadullista ennustamista. Kvalitatiivinen ennustaminen pohjautuu ihmisten käsityksiin asioista sekä järjestelmänvalvojien kollektiiviseen tietoon. Kvalitatiivinen ennustaminen antaa usein vihjeitä suunnitteluun ja on apuna kvantitatiivisen ennustamisen lisäksi hyvän ennusteen tuottamisessa. (Makridakis ym. 1998, 3 - 12.)

Kvantitatiivisessa eli määrämuotoisessa ennustamisessa on viisi vaihetta. Ensimmäinen vaihe on ongelman määrittäminen, joka on usein vaikein vaihe ennustamisessa. Se edellyttää syvää tietämystä siitä, miten ennusteita tullaan käyttämään, kuka ennusteita käyttää ja kuinka ennusteet sopivat kohdeorganisaation käyttöön. Ennusteita tehtäessä on erityisen hyödyllistä jutella ihmisten kanssa, jotka keräävät dataa ja tulevat käyttämään laskettuja ennusteita. Toinen vaihe on datan kerääminen. Tyypillisimmin kerätään kahdenlaista dataa. Tilastollista dataa, joka on usein numeerista, sekä kollektiivista tietämystä ja asiantuntemusta. On tärkeää kerätä dataa historiadataksi, esimerkiksi tietokantaan, jotta valitun ennustemallin avulla voidaan laskea ennusteita. Kolmas vaihe on alustava analyysi. Analyysin avulla saadaan tietää mitä kerätystä datasta voi saada selville. Tällaisia asioita ovat esimerkiksi dataa kuvaavaa tilastotietoa kuten keskiarvo, mediaani, keskihajonta, minimi ja maksimi sekä muita tunnusmerkkejä. Neljäs vaihe on ennustemallin valinta. Ennustemalli valitaan kerätyn datan ja halutun ennustetarkkuuden mukaan kvantitatiivisten ennustemallien joukosta, jotka tuottavat määrämuotoisen ennusteen. Viides ja viimeinen vaihe on ennustemallin käyttö ja arviointi. Kun en-

nustemalli on valittu ja sen parametrit määritetty, voidaan ennustemallia käyttää ennusteiden laskemiseen. Laskettujen ennusteiden käyttäjien tulisi arvioida laskettujen ennusteiden hyvät ja huonot puolet pidemmällä aikavälillä, jotta ennusteita ymmärrettäisiin paremmin. Laskettujen ennusteiden virheiden määrää on myös hyvä arvioida. Näin ennustemallia voidaan uudelleen parametroida vastaamaan tarpeita tai tarvittaessa kokonaan vaihtaa. (Makridakis ym. 1998, 13 - 15.)

Aikasarjaennustamisessa pyritään ennustamaan aikasarjan tulevia arvoja historia-arvojen pohjalta. Aikasarjaennustamista voi toteuttaa usealla eri menetelmällä. Näitä ovat aikasarja trendillä, aikasarja trendillä ja kausivaihtelulla sekä aikasarja ilman systemaattista vaihtelua. (Taanila 2011, 8.) Aikasarja ilman systemaattista vaihtelua on näistä helpoin ymmärtää ja antaa riittävän ymmärryksen miten ennustamista toteutetaan.

Aikasarjadataa ilman systemaattista vaihtelua voidaan ennustaa liukuvan keskiarvon menetelmällä tai eksponentiaalisen tasoituksen menetelmällä. Ennustemenetelmä valitaan siten, että ennusteiden keskimääräinen virhe on mahdollisimman pieni. Hyvän ennustemenetelmän avulla systemaattinen vaihtelu saadaan ennustettua, ennustevirhe kuvastaa tällöin satunnaisvaihtelun osuutta. Ennustevirhe voidaan jälkikäteen laskea vähentämällä ennusteesta toteutunut kysyntä. Useamman ennusteen perusteella voidaan laskea keskimääräinen ennustevirhe. Ennustevirheen mittaamisessa usein käytettyjä mittareita ovat virheiden itseisarvojen keskiarvo (MAD, Mean Absolute Deviation) sekä virheiden neliöiden keskiarvo (MSE, Mean Squared Error). Keskimääräinen ennustevirhe kuvaa kuinka suuriin ennustevirheisiin täytyy keskimäärin varautua. Jos keskimääräinen ennustevirhe kasvaa, on syytä tarkistaa ennustemenetelmän toimivuus. (Taanila 2011, 8.)

Liukuvan keskiarvon menetelmän avulla ennuste lasketaan toteutuneiden havaintojen keskiarvona. Tällöin on päätettävä kuinka usean havainnon keskiarvoa käytetään laskennassa. Mitä useamman havainnon keskiarvoa käytetään, sitä enemmän aikasarjassa esiintyvää satunnaista vaihtelua tasoitetaan. Jos uudempiä havaintoja halutaan painottaa enemmän kuin vanhoja, keskiarvo voidaan laskea painotettuna keskiarvona. Valintaa tehdessä auttaa ennustevirhe, jota yritetään saada mahdollisimman pieneksi. (Taanila 2011, 2 - 3.)

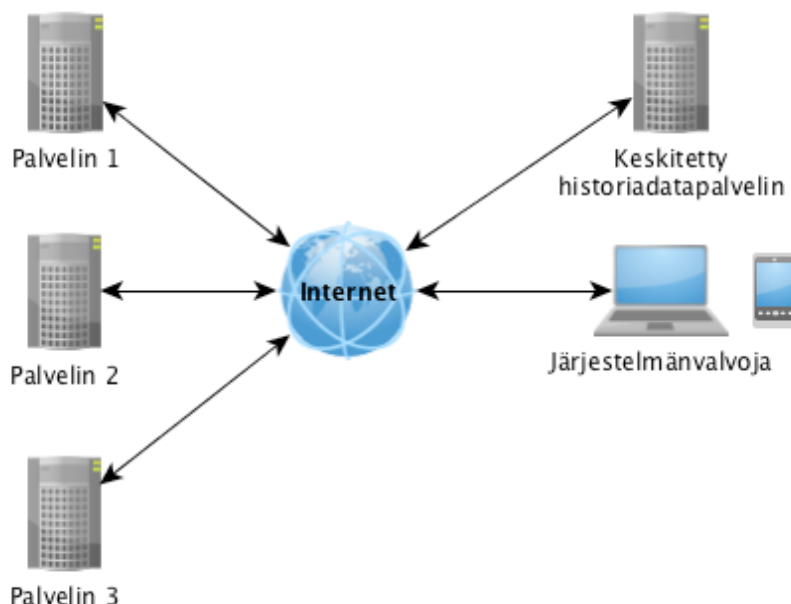
Eksponentiaalisen tasoituksen menetelmää voidaan vaihtoehtoisesti käyttää painotetun keskiarvon sijaan. Ennuste lasketaan tällöin kaavalla $\alpha \cdot \text{edellinen havainto} + (1 - \alpha) \cdot \text{edellinen ennuste}$. Ideana on saada ennuste viimeisen havainnon ja siihen liittyneen ennusteen painotettuna summana. Painokerroin α on luku nollan ja yhden väliltä, joka ilmaisee kuinka suurella painolla edellistä havaintoa painotetaan ennusteen laskennassa. α ollessa 0, ennuste on sama kuin edellinen ennuste. α ollessa 1 ennuste on sama kuin edellinen toteutunut havainto. Suuri α arvo antaa ennusteita, jotka reagoivat herkästi aikasarjassa esiintyvään

vaihteluun. Pienet alfan arvot tasoittavat voimakkaasti aikasarjan vaihtelua. Alfa-arvo valitaan yleensä siten, että keskimääräinen ennustevirhe saadaan mahdollisimman pieneksi. (Taanila 2011, 3.)

3 Palvelinvalvonnan nykytilanne yrityksessä

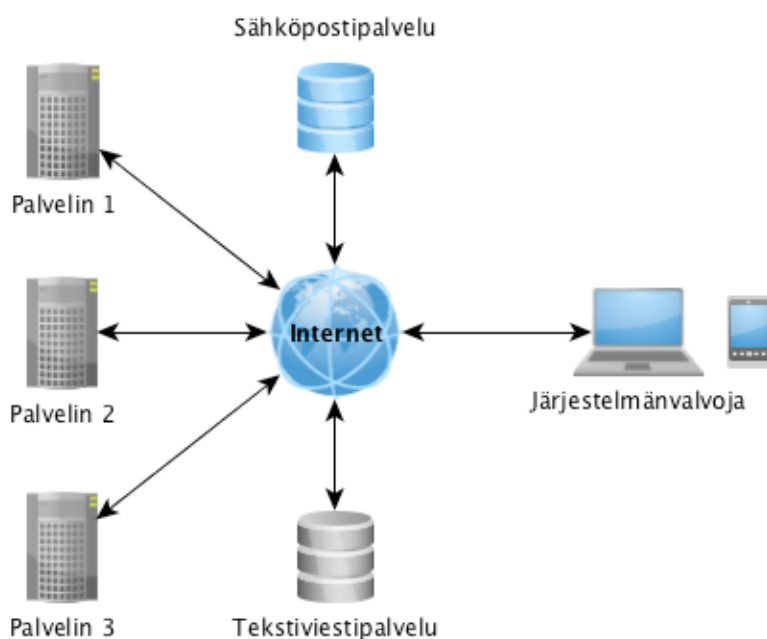
RELEX Oy:n palvelinvalvonnan komponentteja ja rakennetta kuvataan, jotta ymmärtää miten nykyinen palvelinvalvonta toimii ja miten opinnäytetyön tulokset kehittävät sitä. Nykyinen palvelinvalvonta RELEX Oy:n itse kehittämä ja se on toteutettu kahdella eri menetelmällä, jotka vastaavat eri tarpeisiin.

Ensimmäinen menetelmä vastaa kerätyn tilannetietojen historian tallennukseen sekä esitykseen. Tilannetiedot kerätään (pull) periodisesti valvottavilta palvelimilta SSH-yhteyden avulla ja tallennetaan aikasarjadatan tallentamiseen kehitettyyn RRD-tietokantaan keskitetylle historiadatapalvelimelle. Keskitetyn historiadatapalvelimen tietokantaan tallennettua dataa voidaan katsoa siihen tarkoitukseen kehitetyn verkkosivun avulla, jossa tallennettua dataa voidaan katsoa kuvaajina. Tämän tavan hyvä puoli on, että palvelimille ei tarvitse asentaa valvontasovellus-agenttia. Huonona puolena on, että tilannetietojen haku avaa useita peräkkäisiä SSH-yhteyksiä ja siten kuormittaa historiadatapalvelimen sekä valvottavan palvelimen verkkoyhteyttä sekä suorituskykyä. Ensimmäistä menetelmää on havainnollistettu kuvassa 4.



Kuva 4: Tilannetietojen haku (pull) keskitetyn historiadatapalvelimen toimesta.

Toinen menetelmä vastaa reaktiivisen ja proaktiivisen valvonnan tarpeeseen. Reaktiiviseen ja proaktiiviseen valvontaan on kehitetty valvottavalle palvelimelle asennettava valvontasovellus-agentti. Valvontasovellus-agentti lähettää (push) sähköpostin tai tekstiviestin järjestelmänvalvojille kun se havaitsee virheen tai kynnyksarvon ylityksen. Valvontasovellus-agentti myös tallentaa tilannetietoja myöhempää tarkastelua varten, mutta tilannetietoihin ei ole kehitetty käyttöliittymää kuten ensimmäisessä tavassa. Siitä syystä valvontasovellus-agentin tuottamien tilannetietojen visualisointi on käsin tehtävää työtä. Reaktiivisen ja proaktiivisen valvonnan menetelmää on havainnollistettu kuvassa 5.



Kuva 5: Tilannetietojen lähetys (push) valvontasovellus-agentin toimesta.

RELEX Oy:ssä on koettu, että palvelinvalvonta kaipaava kehitystä pystyäkseen vastaamaan yhä vaativampaan palvelinvalvontatarpeeseen. Kehityksen tarve on lähtenyt käytännön ongelmista kuten kerättävän valvontatiedon puutteista sekä valvontasovelluksen ylläpidon haasteista. Täysin selvää sen sijaan ei ole, että minkälaisista palvelinvalvonnan kehitystä kaivataan. Käytössä olevien kahden eri palvelinvalvontasovelluksen ongelmana on, että ne ovat toisistaan erillisiä sovelluksia ja siten vaativat tahollaan ylläpito- sekä kehitystyötä. Ohjelmoinnissa on yleisesti viisasta olla toistamatta tietoa, koska tiedon muuttuessa se pitää muuttaa useaan eri paikkaan. Tätä kutsutaan älä toista itseäsi -toimintaperiaatteeksi. Toimintaperiaatteen idea on vähentää ristiriitoja ohjelmalogiikassa. (Hunt & Thomas 2012, 26 - 27.) Koska palvelinvalvontajärjestelmät mittaavat osin eri asioita, saattaa tulla tilanne jossa tietoa on haettava eri järjestelmistä saadakseen tilanteesta tarpeeksi kattavan kokonaiskuvan. On myös koettu, että tallennetun tilannetiedon esitystavat eivät ole tarpeeksi monipuolisia. Yhdessä näkymäs-

sä pitäisi pystyä saamaan kokonaiskuva valvottavaan palvelimeen, sekä kuvaajien aikaresoluutiota pitäisi pystyä muuttamaan. Järjestelmäresurssien ennustamisen tarve on myös huomattu. Hyväksi on koettu kummankin menetelmän toimintavarmuus. Valvontamenetelmät ovat rajoittuneita, mutta toimivat rajoitteiden puitteissa luotettavasti.

4 Tutkimuksen kuvaus

Tämä luku käsittelee keskeisimmät tieteelliset menetelmät, eli metodologiat, joita opinnäytetyössä käytetään. Luvussa käsitellään myös tutkimuksen teema sekä esitellään tutkimuskysymykset, nämä kuvaavat tarkemmin mitä ja ketä tutkitaan ja missä asiayhteydessä. Opinnäytetyössä käytettyjen tutkimusmenetelmien kuvaus -luku kertoo mitä tutkimusmenetelmiä on käytetty ja miksi niihin on päädytty. Lisäksi verrataan valittuja tutkimusmenetelmiä muihin tutkimusmenetelmiin. Datan kuvaus ja datan kerääminen -luku kertoo mitä dataa opinnäytetyössä analysoidaan, jotta esitettyihin tutkimuskysymyksiin voidaan vastata ja miten datan keräys on tehty. Analysoinnin tehtävänä on kertoa kuinka kerätystä datasta syntyy informaatiota ja lopulta tietoa ja miten saadusta tiedosta johdetaan tietojärjestelmien kehitysehdotuksia. Tutkimuksen kulku kuvaa opinnäytetyöhön tehdyt tapaustutkimukset, tapaustutkimusten tutkimusmenetelmät, datan, datan analysoinnin sekä tulokset.

4.1 Tutkimuksen teema ja tutkimuskysymykset

Opinnäytetyön teemana on palvelinvalvontasovellukset, jotka valvovat ja raportoivat järjestelmän tilaa järjestelmänvalvojille. Opinnäytetyö tutkii palvelinvalvonnan merkitystä liiketoiminnan ja palvelinylläpidon näkökulmasta sekä vertailee nykyisen ja kahdeksan tunnetun palvelinvalvontasovelluksien soveltuvuutta yrityksen RELEX Oy tarpeisiin antaen ehdotuksen palvelinvalvontatoiminnan kehitykseen yrityksessä. Opinnäytetyön tutkimuskysymykset ovat

1. TK1: miksi palvelinvalvonta on tärkeä osa järjestelmänvalvontaa ja liiketoimintaa?
2. TK2: miten palvelinvalvontaa voidaan kehittää?

Tämän lisäksi opinnäytetyössä on kaksi tapaustutkimusta, joissa on yhteensä neljä tutkimuskysymystä lisää. Ensimmäinen tapaustutkimus tutkii omana tuotantona kehitetyn valvontasovelluksen käyttöönottoa yrityksessä RELEX Oy. Tapaustutkimuksessa vastataan seuraaviin tutkimuskysymyksiin:

1. TT1.TK1: miten omana tuotantona kehitetyn valvontasovelluksen käyttöönotto on vaikuttanut yrityksessä RELEX Oy päivittäiseen valvontatoimintaan?
2. TT1.TK2: miten valvontasovellus hyödyttää RELEX Oy:n asiakkaita järjestelmänvalvojien näkökulmasta?

Toinen tapaustutkimus tutki kahdeksan palvelinvalvontasovelluksen ja nykyisten käytössä olevien palvelinvalvontasovellusten soveltuvuutta RELEX Oy:n tarpeisiin. Tapaustutkimuksessa vastataan seuraaviin tutkimuskysymyksiin

3. TT2.TK1: miten nykyinen palvelinvalvonta vastaa yrityksen tarpeisiin?
4. TT2.TK2: miten yrityksen palvelinvalvonta kehittyisi palvelinvalvontasovellusta vaihtamalla?

Tapaustutkimusten tuomien tulosten sekä palvelinvalvontasovelluksien tutkimisen myötä saatiin edellytykset vastata opinnäytetyössä esitettyihin tutkimuskysymyksiin TK1 ja TK2. Opinnäytetyön tulokset antavat konkreettisen palvelinvalvontatoiminnan kehitysehdotuksen sekä selkeyttää palvelinvalvontatoiminnan merkitystä RELEX Oy:lle.

4.2 Käytettyjen tutkimusmenetelmien kuvaus

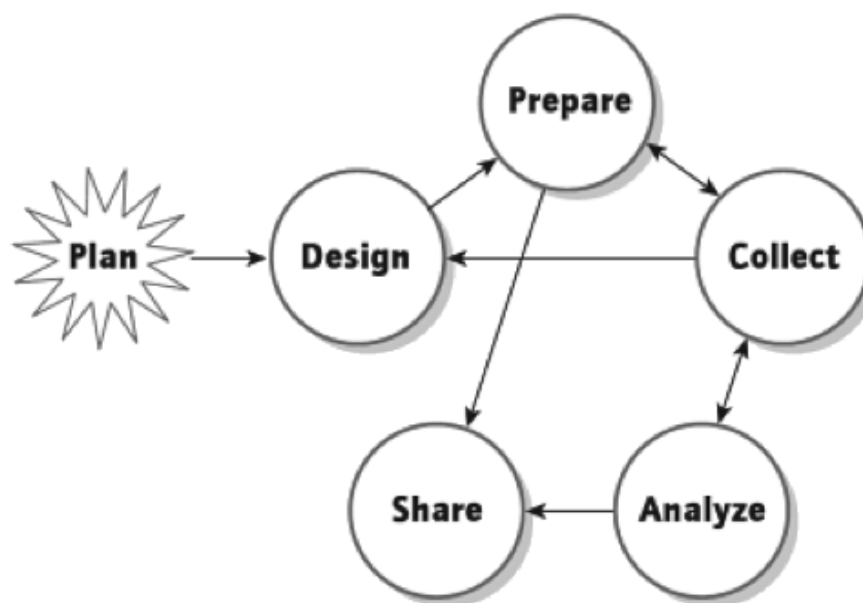
Opinnäytetyön tutkimukset on tehty tapaustutkimuksina. Tapaustutkimus on empiirinen, eli kokemusperäinen tutkimus ja vaatii empiirisiä tutkimusmenetelmiä. Tutkimus perustuu tutkimuskohteen havainnointiin ja mittaamiseen. Opinnäytetyöhön on valittu kaksi empiiristä tutkimusmenetelmää sillä perusteella, että ne tuottavat mahdollisimman paljon tutkimuskysymyksiin kohdennettua tietoa yrityksessä tehtävästä palvelinvalvontatoiminnasta. Tutkimusmenetelmät ovat haastattelu ja havainnointi. Tämän lisäksi opinnäytetyön teoriatausta luukuun on tehty kirjallisuuden koostaminen. Kirjallisuudesta koostettu teoriataustaluku on auttanut opinnäytetyössä tuottamaan teoreettisia lisätietoa aihealueista sekä ongelmanratkaisuehdotuksia.

Monet tutkijat, muun muassa Brannen ja Robson (1992), puhuvat menetelmien yhdistämisen puolesta. Käyttämällä samaan aikaan useampaa tutkimusmenetelmää, saadaan laajempia näkökulmia ja siten voidaan lisätä tutkimuksen reliabiliteettiä eli luotettavuutta. Tutkimuksia on joskus jopa kritisoitu liian yksioikoisista tiedonkeruun tavoista. Hirsjävi & Hurmeen (2001) mukaan Brannen toteaa, että kenttätutkimukset, eli luonnollisissa oloissa tehdyt havaintoaineistojen keräykset, joissa ei yhdistetä esimerkiksi havainnoinnin ja haastattelun tuloksia, nähdään kapeina ja heikkoina tutkimuksina. (Hirsjärvi & Hurme 2001, 38 - 39.) Tutkittavat asiat ovat yleensä suuria kokonaisuuksia ja tutkimukset koostuvat useista eri tutkimusvaiheista. Tämän vuoksi on järkevää hyödyntää useampaa tutkimusmenetelmää, sillä tutkimusmenetelmillä on vahvuutensa sekä heikkoutensa eri tilanteissa. (Mingers 2001, 244.) Edellä mainituista syistä opinnäytetyössä käytetään toisiaan täydentäviä menetelmiä haastattelua, havainnointia sekä teoriaa. Haastatteluiden avulla pyritään saamaan tietoa kohdennettujen kysymysten muodossa. Havainnointi pyrkii tukemaan sekä paikkaamaan haastattelua niiltä osin,

mitä haastatteluissa ei ole huomattu tai osattu vajaiden lähtötietojen vuoksi kysyä. Havainnointi tuo myös hiljaista tietoa. Hiljainen tieto voi olla tiedostamatta tai tiedostetusti sanomatta jäämättä tietoa.

4.2.1 Tapaustutkimuksen vaiheet

Kaikkia tapaustutkimuksia yhdistää se, että ne yrittävät selventää tehtyä päätöstä tai päätöksiä sekä selittää miksi ja miten niitä tehtiin ja mitkä olivat näiden päätösten tulokset. Tästä syystä tapaustutkimuksissa esitetyt tutkimuskysymykset alkavat kysymyksillä miksi tai miten. Tapaustutkimuksen keskeisiä informaatiolähteitä ovat muun muassa dokumentaatio, arkistodata, haastattelut, tapahtumien havainnointi ja osanotto sekä fyysisten tuotoksien tarkastelu. (Yin 2009, 4, 17, 102.) Laadullisten havaintojen lisäksi voidaan tapaustutkimuksessa myös kerätä määrällisiä havaintoja kyselyiden tai aikasarjadatan muodossa. (Eisenhardt 1989, 534 - 535). Tapaustutkimuksen tekijällä ei tulisi olla mahdollisuutta vaikuttaa tutkimuksen aikana tapahtumien kulkuun. (Yin 2009, 4, 17, 102.) Kuvassa 6 on havainnollistettu tapaustutkimuksen iteratiiviset vaiheet.



Kuva 6: Tapaustutkimuksen iteratiiviset vaiheet (Yin 2009, 2).

Tapaustutkimuksen rakentaminen alkaa suunnittelulla (plan), jonka tavoitteena on tunnistaa tutkimusongelma ja asettaa tapaustutkimukselle tutkimuskysymykset. Suunnitteluvaiheessa kannattaa tutustua myös muihin tutkimusmenetelmiin ja punnita niiden vahvuuksia sekä heikkouksia. (Yin 2009, 3; Benbasat, Goldstein & Mead 1987, 374.) Tässä työssä tapaustutkimusten tutkimusongelmat ja tutkimuskysymykset johdettiin työelämän ongelmien sekä tarpeiden poh-

jalta. Tutkimusta oli tehty, jotta tutkimusongelmat ja tutkimuskysymykset saataisiin selvitettyä. Tutkimusmenetelmiin tutustuesssa tutkimusmenetelmien valinta kummankin tapaustutkimuksen osalta oli hyvin selkeä. Menetelmät haastattelu ja havainnointi valittiin tietotarpeiden sekä sopivuutensa puolesta.

Tämän jälkeen tapaustutkimusta luonnostellaan (design). Luonnosteluvaiheessa määritetään tapaustutkimuksen analysointiyksikkö (unit of analysis) ja todennäköinen tapaus tai tapaukset, joita tullaan tutkimaan. Luonnosteluvaiheessa tutustutaan tutkimukseen teoriataustaan ja muodostetaan teoria tutkimuksen kulusta. Näiden avulla saadaan luonnos tutkimuksesta. (Yin 2009, 25; Benbasat ym. 1987, 372.) Tässä työssä tutkimusongelman pohjalta oli mahdollista määrittää tutkittavat tapaukset alusta alkaen tarkasti. Tehtyjen luonnosten pohjalta tutkijalle hahmottui tapaustutkimuksen kulku entistä selkeämmin.

Tapaustutkimuksen edetessä valmisteluvaiheeseen (prepare) tutkijan tulisi kerrata ja omaksumaa tapaustutkimuksen kulku ja menetelmät. Tutkijan tulisi myös perehtyä tutkittavaan aiheeseen huolellisesti ja muodostaa koetapaus. Tässä vaiheessa on myös hyvä saada hyväksyntä tutkimukselle ja tarvittaessa käytettävälle menetelmille tutkimuksen sidosryhmiltä, eli tahoilta joiden kanssa opinnäytetyö on tekemisissä. (Yin 2009, 66; Benbasat ym. 1987, 370.) Luonnostelun, valmistelun ja menetelmien kertauksen pohjalta oli helppo saada tutkimuksille suunta, jota voitiin tutkia syvemmin. Kummassakin tapaustutkimuksessa suunta tuntui heti alkuun luonnolliselta sekä oikealta. Tutkimuksen kiinnostaessa tutkijaa, tuntui tutkimuksen teko miellyttävältä. Lisäksi hyväksyntä tutkimukselle oli saatu.

Datankeruuvaiheessa (collect) tapaustutkimuksessa kerätään nimensä mukaisesti tapaustutkimukselle tärkeää dataa esimerkiksi tutkimalla arkisto-dataa, dokumentteja ja tutkittavaa kohdetta, tekemällä haastatteluita sekä havainnointia. Kerätyn datan avulla voidaan muodostaa tapaustutkimukselle asiatietokanta, jonka avulla tutkimuksen analyysia voidaan johtaa. Useiden eri tutkimusvaiheiden vuoksi dataa kertyy vähitellen. Eri tutkimusvaiheiden vuoksi on hyvä käyttää niihin soveltuvia tutkimusmenetelmiä. (Yin 2009, 98; Mingers 2001, 244; Benbasat ym. 1987, 374.) Tässä työssä tapaustutkimuksen dataa kerättiin koko tapaustutkimuksen elinkaaren ajan usean tutkimusmenetelmän kehikon periaatteen mukaisesti. Dataa kerättiin tutkimusmenetelmin ja se taltioitiin pääasiassa Microsoft Excel -taulukoihin sekä Microsoft Word -tekstidokumentteihin.

Datan analysointi vaihe (analyze) koostuu datan tutkimisesta, kategorisoinnista, taulukoinnista, testauksesta ja datan yhdistämisestä. Kerätyn datan avulla voidaan luoda johtopäätöksiä analysoidusta aineistosta. Analysointivaihe on usein hankalin vaihe tapaustutkimuksessa, sillä kerätty data on usein monimuotoista. Data voi olla kvalitatiivista tai kvantitatiivista ja kerätty asiatietokantaan tai yksittäisinä dokumentteina. Analysoinnin tekoon on monta erilaista ta-

paa. Suositeltavaa on järjestää dataa taulukkoihin kategorioittain, luoda kuvaajia tai muuta grafiikkaa, mitata tapahtumien toistuvuutta sekä luoda numeerisia tunnusmerkkejä kuten esiintymien keskiarvoa tai keskihajontaa. Usein myös datan pistäminen kronologiseen järjestykseen auttaa tekemään johtopäätöksiä. Lisäksi todistusaineisto johtopäätösten johtamiseen tulisi myös esittää. (Yin 2009, 126 - 129; Benbasat ym 1987, 374.) Tässä työssä datan analysointia on tehty vuorovaikutteisen analysointiotteen menetelmällä, joka on esitetty Miles, Huberman & Saldañan *Qualitative Data Analysis* kirjassa (2014). Tehdyn analyysin avulla voitiin johtaa tutkimuksen tulokset.

Viimeinen vaihe tapaustutkimuksessa on raportointi (share). Tapaustutkimuksen raportoinnilla tarkoitetaan tulosten ja löydösten saattamista päätökseen. Raportin muodosta riippumatta on tärkeitä tunnistaa lukijakunta, luoda aineistosta tutkimuksen rakenne sekä antaa raportin vedoksia luettavaksi katselmoijille. Raporttia on hyvä työstää iteratiivisen periaatteen mukaisesti koko tapaustutkimuksen ajan. Raportissa olevan kirjallisuuteen nojaavan teoriataustan sekä metodologian voi luoda jo varhaisessa vaiheessa tapaustutkimusta. Raportissa on hyvä olla tarpeeksi lukijalle todistusaineistoa, jotta hän voi luoda omat johtopäätöksensä. Lopullinen raportti koostuu tekstistä sekä visuaalisesta aineistosta, joka katselmoidaan useamman ihmisen toimesta. Raportti kirjoitetaan korjaamista vaativilta osin uudelleen, kunnes lopputulos on haluttu. (Yin 2009, 164.) Tapaustutkimuksien raportointi esitettiin tehdyn analyysin pohjalta mahdollisimman totuudenmukaisesti. Raportteja kehitettiin koko tutkimuksen elinkaaren ajan, mutta varsinkin loppupuolella ja katselmoinnin myötä ne paranivat huomattavasti. Tutkimusraportit sisälsivät lukijalle tärkeän todistusaineiston, jotta esitettyjen asioiden totuudenmukaisuus pystyttiin todentamaan. Kummankin tutkimusraportin tulokset -luku tuotiin tähän opinnäytetyöhön tutkimuksen kulku -lukuun, sekä käytettiin tutkimuksen datana.

Opinnäytetyöhön on tehty kaksi tapaustutkimusta. Tapaustutkimusten tarkoituksena on tuottaa kohdennettua tietoa yrityksessä tapahtuvasta palvelinvalvontatoiminnasta sekä palvelinvalvonnan merkityksestä. Tapaustutkimusten tuottamaa tietoa käytetään vastaamaan opinnäytetyössä esitettyihin tutkimuskysymyksiin. Tapaustutkimukset on tehty kirjan *Case Study Research Design and Methods* (Yin 2009) ohjeistuksen mukaisesti käyden läpi tapaustutkimuksen iteratiiviset vaiheet. Opinnäytetyössä esitetään tulokset luku tehdyistä tapaustutkimusten raporteista.

4.2.2 Haastattelu

Haastattelu on yksi käytetyimmistä tiedonkeruumuodoista ja se on tiedonhankinnan perusmuoto. Joustavana menetelmänä haastattelu sopii moniin erilaisiin tutkimuksen tarkoituksiin. Haastattelussa ollaan suoraan kielellisessä vuorovaikutuksessa tutkittavan asian kanssa, joka luo mahdollisuuden suunnata tiedonhankintaa tilanteessa itsessään. Samalla saadaan tietoa

vastausten takana olevista motiiveista. Haastattelu sopii erityisesti silloin, kun halutaan selvittää vastauksia ja syventää saatavia tietoja esimerkiksi pyytämällä mielipiteitä ja perusteluja saatuihin vastauksiin. Haastattelu sopii erityisesti myös silloin, kun kyseessä on vähän kartoitettu, tuntematon alue. (Hirsjärvi & Hurme 2001, 34 - 35.) Haastattelu toimii havainnoinnin tukena, havainnoitaessa on pakko valita mitä asioita tai ihmisiä havainnoidaan ja minä ajankohtana, mutta haastattelu yltää myös tilanteisiin, joita havainnoitsija ei saavuta. Haastattelun avulla saadaan myös käsitys miten juuri haastateltavat tulkitsevat asioita, jolloin näkemyksiin saadaan tarvittaessa subjektiivinen ote. Haastattelun haasteena on ymmärtää, mitä haastateltava sanoo ja oikein tulkitseminen. Organisaatiot kehittävät usein oman kielensä käsittelemiensä asioiden ympärille, jota saattaa ulkopuolisen haastattelijan olla vaikea ymmärtää. On siis tärkeää tutustua aihealueeseen sekä termistöön ennalta ja kysyä selvittäviä kysymyksiä haastattelutilanteessa. (Patton 1990, 179; Walsham 2006, 323; Becker & Geer 1957, 29.)

Haastattelussa tutkijan tavoitteena on muodostaa mielikuva haastateltavan henkilön kokemuksista, ajatuksista, tunteista ja käsityksistä. Haastattelu on tiedon keräämisen metodi, jota ohjaa haastattelija. Haastattelu on myös päämääräohjattua ja ennalta suunniteltua. Hyvin suunniteltu haastattelu koostuu osista, jotka ovat esittely, lämmittely, varsinainen haastattelu, yhteenveto kysymykset ja kiitokset. Haastattelun avulla voidaan käsitellä nykyisyyttä, tulevaisuutta sekä menneisyyttä. (Hirsjärvi & Hurme 2001, 38 - 43; Hirsjärvi, Hurme & Sajavaara 2001, 204 - 206; Walsham 2006, 321.) Haastattelujen luottamuksellisuuden säilyttäminen ja sen tuominen esille haastateltavan kanssa heti haastattelun alussa on tärkeitä, jotta haastateltava antaisi mahdollisimman totuudenmukaisia vastauksia sekä uskaltautuu puhumaan mahdollisimman totuudenmukaisesti. Haastattelun tulosten laadulle on myös tärkeää, että haastattelut tehdään haastateltavalle sopivana ajankohtana ja että haastattelut eivät veisi liikaa aikaa, jotta haastateltava ei tuntisi oloaan painostetuksi. (Walsham 2006, 321 - 322.)

Haastattelu on tutkimusmenetelmänä tämän opinnäytetyön reliabiliteetin, validiteetin ja tiedon saamisen kannalta erittäin tärkeä. Haastattelu antaa tutkimukselle näkökulmia joita ei muilla tutkimusmenetelmillä saa. Haastattelu tutkimusmenetelmänä takaa, että tutkimuksessa esiintyvät asiat eivät ole vain tutkimuksen tekijän näkemyksiä asioista. Haastattelun avulla saadaan tärkeää tietoa nykyisestä käytössä olevasta palvelinvalvonnasta. Nykyinen palvelinvalvonta yrityksessä on itse kehitetty ja sen vuoksi siitä ei ole saatavilla ulkopuolista dokumentaatiota tai tutkimusta. Haastattelujen avulla selvitetään käyttäjien vaatimuksia uudelle palvelinvalvontasovellukselle sekä saadaan tietoa nykyisen palvelinvalvontasovelluksen järjestelmälliseen ja kattavaan analysointiin. Haastattelujen avulla myös kartoitetaan miten ihmiset kokevat palvelinvalvontasovelluksen vastaavan heidän tarpeisiinsa ja onko nykyisin käytössä olevassa palvelinvalvontasovelluksessa selkeitä virheitä tai puutoksia. Haastattelu kohdis-

tetaan neljään tekniseen henkilöön, jotka ovat palvelinvalvontasovelluksen pääasiallisia käyttäjiä. Haastattelut äänitetään haastateltavien luvalla, kestoaltaan haastattelut ovat 15 - 30 minuuttia pitkiä. Haastattelut litteroidaan haastatteluiden jälkeen ja analysoidaan opinnäytetyön tekijän toimesta johtaen siitä yhden yhtenäisen dokumentin.

Haastattelukertoja on suunniteltu kaksi kappaletta, ensimmäinen haastattelukerta tehdään opinnäytetyön aikana, jotta voidaan ymmärtää palvelinvalvonnan nykytilaa. Toinen suunniteltu haastattelukerta ajoittuu opinnäytetyön jälkeiseen aikaan, kun palvelinvalvonnan kehitystä on saatu aikaiseksi. Toisen haastattelukerran tavoitteena on iteroiden kehittää palvelinvalvontaa vastaamaan tarpeita vieläkin paremmin ja sen suorittaminen esitetään opinnäytetyössä jatkokehitysehdotuksena. Haastattelu olisi hyvä tehdä vähintään neljälle henkilölle, joka osoittautui riittäväksi määräksi ensimmäisellä haastattelukierroksella.

Haastattelu	Kuvaus	Haastattelujen määrä (n)	Suunniteltujen kysymysten määrä (n)
Haastattelut #1	Palvelinvalvonnan nykytila, liite 1	4	7
Haastattelut #2	Palvelinvalvonnan tila tehdyn kehityksen jälkeen, liite 2	≥ 4	7

Taulukko 1: Suunnitellut haastattelut.

4.2.3 Havainnointi

Havainnointi on välttämätön perusmenetelmä, joka ulottuu kaikille tieteenaloille. On väitetty, että tieteellinen tieto perustuu havaintoihin, jotka on tehty todellisuudessa. Havainnointia voidaan käyttää tarkkailtaessa sekä käyttäytymistä, että kielellistä ilmaisua. Havainnoitava kohde voi olla tietoinen tai tietämätön havainnoinnista. Tieteellinen havainnointi vaihtelee informaalista, toisin sanoen vapaamuotoisesta täysin systemaattiseen ja kontrolloituun havaitsemiseen. Havainnointia voidaan käyttää yhdessä muiden tiedonkeruumenetelmien kuten haastattelun kanssa. (Hirsjärvi & Hurme 2001, 37 - 38.) Havainnointia käytetään usein kun tutkimuksen kohteesta tiedetään verrattain vähän ja halutaan saada yleistä tietoa. Sen avulla voi muodostaa tutkimukselle tärkeitä hypoteeseja ja saada tutkimuksen loppuvaiheessa myös tärkeitä tietoa kohdentamalla havainnointia. (Nunamaker ym. 1991, 95.)

Havainnointi on menettelynä erittäin vaativa. Parhaiden tuloksien saamiseksi se edellyttää tutkijalta huolellista koulutusta, ellei havainnointia taltioida esimerkiksi videokameran avulla. Havainnoitava tapahtuma voi olla niin suuri, että kokemattoman havainnoitsijan on vaikea kirjata kaikkea tutkimukselle tärkeää tapahtumaa. Myös tapahtumat ovat usein ainutkertaisia. Havainnoinnista saatava tieto koskee yksinomaan nykyisyyttä, sillä ei voida saada tietoa

esimerkiksi menneisyydestä. (Hirsjärvi & Hurme 2001, 38.) Havainnoinnin kirjaus on kuitenkin aina jossakin määrin subjektiivinen havainnoitsijan näkemys tilanteesta, vaikka sitä aktiivisesti yritettäisiin välttää. Tilanteet itsessään tarjoavat aitoja havaintoja, joten havainnoitsijalla on tärkeä rooli niiden kirjaamisessa siten, että niistä hyöty tutkimukselle on objektiivinen. (Mingers 2001, 245; Nunamaker ym. 1991, 95.) Havainnoinnin avulla ei kuitenkaan voida kuitenkaan tallentaa kaikkea tietoa, siksi on tarpeellista tehdä valitaan mitä halutaan havainnoida ja mihin ihmisiin havainnointi kohdistuu. Tämän lisäksi havainnointi tapahtuu ennalta suunnitellulla ajanjaksolla. (Patton 1990, 179.) Havainnointi auttaa myös ymmärtämään mitä tietoa tutkimukselta puuttuu ja pystyvätkö tutkimuksessa käytetyt tiedonkeruumenetelmän tuottamaan tätä tietoa. Havainnointi myös tuo usein viihkeitä kysymyksistä, joita haastattelussa kannattaa tutkia tarkemmin. (Becker & Geer 1957, 28.)

Havainnointi on haastattelun ohella opinnäytetyön tutkimusmenetelmien toisessa pääroolissa. Havainnoinnin tarkoituksena on paikata kaikkea sitä, mitä haastatteluissa ei tullut ilmi. Palvelinvalvontasovelluksen käyttökokemukseen liittyvät hyvät, huonot tai kehitysluontoiset asiat tulevat usein esille vasta kun jotakin tapahtuu ja palvelinvalvontasovellusta käytetään ratkaisemaan aito tilanne. Opinnäytetyössä tehtävään tutkimukseen osallistuvia henkilöitä havainnoidaan normaalien työtehtävien ohella, sitä mukaa kun palvelinvalvontasovellukseen liittyviä seikkoja tulee ilmi. Varsinaista havainnointiaikataulua ei ole, vaan havainnointia toteutetaan koko opinnäytetyön ajan.

Opinnäytetyön tekijän aiempi kokemus havainnoinnista on osoittanut, että kyseisellä tutkimusmenetelmällä saadaan usein monenlaista tietoa jota muuten ei saataisi. Joskus haastateltava ei halua kertoa kaikkea. Tällaisia tilanteita saattaa esiintyä kun jotakin asiaa ei osata tehdä ja siten ei kehdeta myöntää. Havainnointi usein myös tuo esille asioita, joita tarkasteltava henkilö pitää vähäpätöisinä, mutta jotka todellisuudessa saattavat olla hyvinkin merkittäviä tutkimustulosten kannalta. Havainnointi myös kertoo toimivatko ihmiset kuten he haastatteluissa sanovat toimivansa. Tämä tukee valittujen tutkimusmenetelmien haastattelu ja havainnointi käyttöä yhteistyössä.

4.3 Datan kuvaus ja datan kerääminen

Datan ja sen keräyksen selkeä kuvaus, datan lähteet sekä näiden merkittävyys työlle on tärkeä osa tutkimuksen reliabiliteetin sekä validiteetin osoitusta. (Benbasat ym. 1987, 381; Dubé & Paré 2003, 612.)

Opinnäytetyössä datan keruu on tehty tutkimusmenetelmien haastattelu sekä havainnoinnin avulla, joita on käytetty tapaustutkimuksissa 1 ja 2. Haastattelut sekä havainnointi on tehty kohderyhmälle (n=4), joka on RELEX Oy:n palvelinvalvontasovelluksia käyttävä henkilöstö.

Tutkimusmenetelmien (n=2) avulla on kerätty dataa siitä, miten yritys käyttää palvelinvalvontasovellusta ja mitä vaatimuksia palvelinvalvontasovellukselle esiintyy. Tämän lisäksi palvelinvalvontasovelluksien testaus sekä teorian tiedon läpikäynti on antanut arvokasta tietoa siitä mihin palvelinvalvontasovellukset pystyvät. Näiden tietojen avulla on voitu arvioida voidaanko haastatteluiden sekä havainnoinnin avulla ilmenneitä asioita ratkaista tavoilla, jotka ovat ilmenneet teorian ja testauksen avulla. Tämän lisäksi dataa kerääntyy, kun dataa tiivistetään sekä havainnollistetaan. Kerätystä sekä havainnollistetusta datasta johdetaan tiivistettyä dataa. Datan esittämistä ja havainnollistamista tapahtuu koko opinnäytetyön elinkaaren ajan. Dataa kirjataan taulukoihin (n=2) tekstimuotoisena muodostaen loogisia kategorioita. Datan esittämistä, tiivistämistä ja kategorisointia käytetään hyväksi, jotta dataa ymmärrettäisiin paremmin ja sen merkitys selkeytyisi.

Havainnollistaminen toimii myös opinnäytetyön tuloksien tukena, auttaen lukijaa luomaan omat johtopäätöksensä havainnollistetusta datasta. Opinnäytetyössä datan tulkitsemista sekä johtopäätöksien syntyä tapahtuu vuorovaikutteisen analysointiotteen mukaisesti kun datan ymmärrys ja merkitys alkaa selkiytyä.

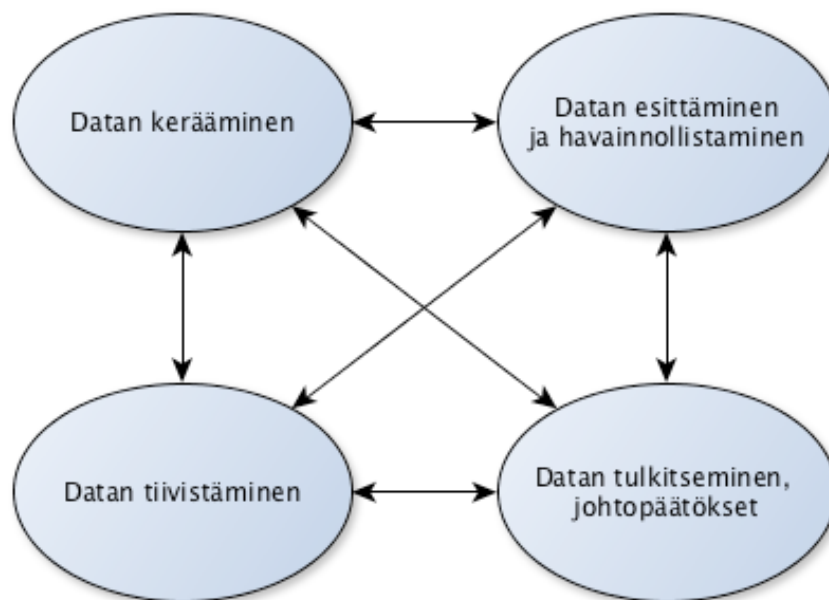
4.4 Analysointi

Tutkimuksen analysointiyksikkö määrittää tutkimuskohteen, jota opinnäytetyössä pääasiassa tutkitaan. Tutkimus tuottaa monenlaista tietoa, mutta analysointiyksikön tehtävänä on selkeyttää mikä on tutkimuksen tärkein tutkimuskohde. (Yin 2009, 31.) Opinnäytetyön analysointiyksikkö on palvelinvalvontasovellus.

Opinnäytetyössä käytettiin *Qualitative Data Analysis* -kirjassa (Miles, Huberman & Saldaña 2014) esitettävää kvalitatiivisen datan vuorovaikutteista analysointimenetelmää, jonka avulla pystyttiin vastaamaan opinnäytetyön aiemmassa luvussa esitettyihin tutkimuskysymyksiin. Kvalitatiivisiin menetelmiin joskus myös viitataan nimillä pehmeät tai laadulliset menetelmät (Eskola & Suoranta 1998, 13). Kvalitatiivisessa tutkimuksessa yritetään vihjeiden ja johtolankojen pohjalta selvittää ilmiöitä. Kvalitatiivisen datan vahvuuksiin kuuluu luonnolliset tapahtumat luonnollisissa ympäristöissä, joka takaa datan pohjautuvan tosielämän tapahtumiin. Tyypillisesti aikajaksolta kerätty kvalitatiivinen data sopii erityisesti prosessin tutkimiseen. Data vastaa eri ajankohtien tilannekuvaa ja vastaa kysymyksiin mitä ja kuinka paljon antaen ymmärryksen kysymyksiin kuinka ja miksi asiat tapahtuvat kuten tapahtuvat. (Miles ym. 2014, 11.)

Qualitative Data Analysis -kirjassa (Miles ym. 2014) esitetty vuorovaikutteinen datan analysointiotte tyypillisesti alkaa kerätyn datan tiivistämisellä. Sen jälkeen se siirtyy aineiston esittämiseen ja havainnollistamiseen. Viimeiseksi päädytään aineiston tulkitsemiseen ja joh-

topäätöksiä. Vuorovaikutteisuudella tarkoitetaan eri datan analysointivaiheiden samanaikaista tapahtumaa ymmärryksen sekä hyvän tiedon luomiseksi. Kuvassa 7 on havainnollistettu datan analysointivaiheita, sekä vaiheiden vuorovaikutteisuutta.



Kuva 7: Vuorovaikutteinen datan analysointiotte (Miles ym. 2014, 40).

Datan keräämisessä käytetään tyypillisesti kahta tai useampaa menetelmää sekä useampia lähteitä, jotta saatu aineisto kuvastaisi mahdollisimman paljon totuutta. Jos dataa on kerätty riittävästi ja se ei ole ristiriidassa keskenään, voidaan datasta tulkita luotettavia johtopäätöksiä. (Benbasat ym. 1987, 347.)

Datan tiivistäminen johtaa usein uusiin ideoihin, joita voidaan käyttää datan esittämisessä ja havainnollistamisessa. Datan lisääminen esittämiseen ja havainnollistamiseen käytettyyn materiaaliin vaatii usein datan uudelleen tiivistämistä. Kun datan esittämiseen ja havainnollistamiseen käytetty materiaali alkaa valmistua, myös alustavat datan tulkinat sekä johtopäätökset syntyvät. Tämän johdosta voidaan esitettävään materiaaliin luoda testiaineistoa, jonka avulla tehdyt johtopäätökset voidaan vahvistaa oikeiksi. (Miles ym. 2014, 14.)

Datan tiivistys on prosessi, jossa kerätystä datasta muodostetaan tutkimukselle tärkeä aineisto karsimalla, yksinkertaistamalla, tiivistämällä ja muokkaamalla. Tiivistäessä datasta ei huku tutkimukselle tärkeää tietoa, vaan merkityksellinen tieto jäsennetään uuteen muotoon, tehden kerätystä aineistosta tutkimukselle merkityksellisempää. (Miles ym. 2014, 12.) Opinnäytetyössä datan tiivistystä tehdään jäsentämällä tapaustutkimusten avulla kerättyä dataa taulukkoon. Eli lähteistä koostettu data tiivistetään ja asetetaan taulukkoon, joka helpottaa ana-

lysoinnin tekemistä. Taulukko, johon on käytetty vuorovaikutteisen datan analysointiotetta löytyy opinnäytetyön liitteestä 3.

Vuorovaikutteisen analysointiotteen mukaisesti datan tiivistystä tapahtuu kvalitatiivisissa tutkimuksissa jo usein tiedostamatta ennen datan keräysvaihetta. Tutkija päättää mitä tutkimuskysymyksiä (n=4) ja datan keräysmenetelmiä (n=2) käytetään. Datan keräyksen edetessä tutkija tiivistää saatua dataa kirjoittaessaan yhteenvetoja, tehden koodituksia, luoden teemoja ja kategorioita sekä tehdessään analyttisiä muistiinpanoja. Datan keräyksen ja muun kenttätöön jälkeen dataa tiivistetään koko tutkimuksen elämänsäajan ajan, kunnes tutkimus on saatu päätökseensä. (Miles ym. 2014, 12.)

Datan tiivistäminen on osa analysointia. Se koostuu tutkijan päätöksistä miten suhtautua dataan. Tutkija tekee analyttisiä päätöksiä, kuten mitä dataa käytetään tutkimuksessa ja mikä pudotetaan pois. Myös miten säilytettävä data kategorisoidaan parhaiten sitä kuvailevaksi ja minkä tarinan data tulee kertomaan. Prosessina datan tiivistäminen selkiyttää, jäsentää, tarkentaa ja hylkää dataa siten, että johtopäätökset ovat muodostettavissa tiivistetyn datan pohjalta. (Miles ym. 2014, 12.)

Alla on ote haastateltavan 1 vastauksesta haastattelussa sekä litteroitu versio. Kokonaisuudessaan haastattelu on kestänyt noin 20 minuuttia. Vastaukset tiivistettiin tutkimukselle merkittäviin osiin siten, että vastaukset ovat keskenään vertailukelpoisia sekä tarvittaessa kvantifioitavissa paremman ymmärryksen saavuttamiseksi.

”T: Onko nykyinen palvelinvalvonta riittävä?

H1: Ei ole. Monitoroinnin signal:noise suhde ei ole kohdallaan. Tietyistä asioista tulee aivan liikaa hälytyksiä ja täten niille turtuu kun toimia ei kuitenkaan tarvitse tehdä sellaisia ovat esimerkiksi Ratio of direct memory to table size -hälytykset. Monitorointi on puhdas point in time valvonta, jossa kerrotaan onko JUURI NYT ongelma, eli osuvatko mittarit hälyrajojen yläpuolelle. Puuttuu trendit ja ennustus. Esimerkiksi lptablesissa voi pakettifiltteröintiä toteuttaa niin että ”blokkaa jos tulee 100 pakettia 1 sekunnissa tai 1000 pakettia 100 sekunnissa” joten voidaan huomata purskeisuus JA kasvanut baseline varoituksissa.”

”T: Miten kehittäisit yrityksen palvelinvalvontaa?

H1: Tärkeimpänä näkisin monitoroitavien mittareiden ennustamisen tulevaan ja arvioinnin ”tämä hälytin tulee laukeamaan 2 päivän päästä ellei sille tehdä mitään”. Tämä siirtäisi työn puhtaasti reaktiivisesta proaktiiviseen suuntaan JA antaa puskuria suunnitella ja priorisoida töitä. Nykymalli jossa on kokoajan tulipaloja ja isoin/viimeisin saa prioriteetin ei ole kestävä tekemistä. Myöskin jokin muu käyttöliittymä kuin sähköposti olisi tarpeen. Olisi järkevää aggregoida mittarivaroittimia esimerkiksi: Palvelin 1, 3 x punainen, 2 x keltainen, 10 x vih-

reä, Palvelin 2 0 x punainen, 12 x keltainen, 5 x vihreä jne. Tämä auttaisi työn priorisoinnissa ja ohjauksessa. Se, miten kategoriat määritellään, pitäisi sopia yhdessä.”

”T: Näkisitkö ongelmaksi kahden palvelinvalvontasovelluksen samanaikaisen käytön?

H1: Kaksi eri softaa voi havaita eri asioita, joten siinä mielessä se ei ole ongelma. Mutta jos sovellukset mittaavat samoja asioita ja antavat eri tuloksia, seuraa siitä vain sekaannusta. Järkevintä olisi kuitenkin jos vanhoista sovelluksista pystytään jotenkin tuomaan data uuteen ja keskittymään yhden sovelluksen kehitykseen ja valvontaan.”

”T: Mitkä mielestäsi ovat nykyisten (Poller, perl scriptit + rrd) valvontasovelluksien vakavimmat puutteet?

H1: perl+rrd - eri palvelimilla on jokaisella oma skriptinsä, joten kun uusi mittari keksitään, se ei ole automaattisesti kaikilla keruussa joten se jää aika usein laittamatta päälle kaikkiin takautuvasti. Poller - uusien mittareiden kehittämisen kankeus - pitää koodata erikseen Javaa, pistää kamat gittiin, deployata kaikkialle ja katsoa että toimii. Ei ole hirveän ketterä tapa.”

”T: Mitä asioita nykyinen palvelinvalvontasovellus (poller) toteuttaa hyvin ja mistä et haluaisi luopua uuden palvelinvalvontasovelluksen myötä?

H1: Poller on vakaa järjestelmä ja tekee ne mittaukset, mitkä siihen on toteutettu, hyvin. Perushyvää suoritusta. En osaa tarkemmin eritellä.”

”T: Nimeä vähintään viisi (5) työssäsi eniten ongelmia aiheuttavaa asiaa, joita on mahdollista valvoa

H1: Kiintolevytila suurella erolla seuraavaan, RELEX sovellus (offheap ja heap -varoitukset, RELEX http-pyyntöjen vasteaika, RELEX yöajojen kestot, CPU-kuorma, prosessien määrä.”

T: Monesti katsot miten asiat ovat menneet historiassa, koetko koskaan tarvetta katsoa miten asiat tulevat menemään tulevaisuudessa? (valvontasovelluskontekstissa eng. prediction, forecasting, trending), minkä mittareiden osalta ennustaminen olisi erityisesti hyödyllistä?

H1: Kyllä ehdottomasti olisi tärkeää ennustaa tulevia ongelmia. Esimerkiksi tietyissä ympäristöissä on havaittavissa selkeä viikkotasoinen trendi (maanantait hitaita ja hidastuvat perjantaihin mennessä) sekä kuukausitasoinen (hidastuu joka kuukausi). Tärkein ennustettava olisi tuo levytila. Se aiheuttaa aina vakavia hälyjä, koska kun levy loppuu kokonaan alkaa asiat mennä pahasti rikki. Levyä kuluu joka palvelimella joka päivä aina vain lisää, mutta siinä saattaa tulla toisinaan isojakin piikkejä ja jos niitä pystyttäisiin ennustamaan, voitaisiin saada kiinni ongelmia itse RELEX-sovelluksessa jne. esimerkiksi eräässä RELEX-asennuksessa normaalisti tietokannan koko kasvaa 20 gigaa päivässä ja eräs päivä se kasvoikin >100 gigaa ja levytila meinasi loppua kertarysäyksellä (nousi >95%:iin käyttö). Tästä ei ollut tullut en-

nen 90% ylitystä mitään infoa. Toinen on nuo muistien loppumiset - mikäli offheapissa on havaittavissa että sen määrä nousee vaikka 1 gigalla viikossa niin voitaisiin helposti ennustaa päivämäärä, johon mennessä tuotekehityksen on pakko saada optimointeja aikaan tai mihin mennessä supportin on yhdessä projektitiimin kanssa pakko tehdä jotain muutoksia esimerkiksi optimointeihin jne. tai RELEX-asennus tulee kuolemaan.”

”T: Sana on vapaa, mitä haluaisit sanoa palvelinvalvonnan kehityksestä?

H1: Kerätylle tiedolle yksi pääte piste, josta voidaan saada hyvä raportointi.”

Litteroidusti ja tiivistetysti haastattelu kääntyy dataksi, kuten nykyinen palvelinvalvonta ei ole riittävä, suurimpia puutteita ovat muun muassa datan trendien, ennustuksen sekä purseiden tunnistaminen. Palvelinvalvontaa tulisi kehittää lisäämällä tapahtumien ennustamisen mahdollisuus sekä dashboard -toiminnallisuus. Usean valvontasovelluksen käyttö ei ole ongelma, jos niillä seurataan eri asioita. Hallinnan dynaamisuuden puute, valvonta on liian staattista. Nykyiset olemassa olevat mittarit toimivat luotettavasti. Suurin osa valvontasovelluksen havaitsemista palvelimen ongelmista liittyy: kiintolevytilaan, RELEX sovellukseen, RELEXin HTTP-sivupyyntöjen vasteaikaan, suoritinkäyttöön ja prosessimäärään. Kiintolevytilan ja sovelluksen resurssikäytön ennusteet sekä trendit tulisi olla saatavilla. Kaikki kerätty tieto tulisi olla tallennettu yhteen tietokantaan.

On hyvä ymmärtää, että datan tiivistäminen ei välttämättä tarkoita datan kvantifioitua, eli asian määrittämistä lukuina tai suureina. Kvalitatiivinen data voidaan tiivistää monin eri tavoin, valitsemalla osan, tiivistämällä tai käyttäen kiertoilmaisuja. Jos datalla on suuri vaikutus tutkimuksessa, on yksityiskohtainen data hyvä muuttaa ylemmälle eli yleistävälle tasolle, jos mahdollista. Ylemmän tason kvalitatiivista dataa on helpompi ymmärtää sekä analysoida. (Miles ym. 2014, 12.)

Datan esittäminen ja havainnollistaminen on jäsenneilyn ja tiivistetyn tiedon rakentamista. Se mahdollistaa datan tulkitsemisen ja johtopäätösten teon. Arkielämän esimerkkinä datan esitys ja havainnollistaminen esiintyy mittareina auton mittaristossa, Facebook-päivityksinä ja sanomalehtinä. Ne auttavat meitä ymmärtämään, mitä on tapahtumassa ja mahdollistavat jatkotutkimisen tai toimenpiteisiin ryhtymisen asian pohjalta. (Miles ym. 2014, 12 - 13.)

Yleisin kvalitatiivisen datan esityksen ja havainnollistamisen muoto on pitkä teksti. Yli tuhat sivun tekstiaineisto on hankalalukuista, hajanaista sekä peräkkäistä. Se on myös huonosti jäsenneiltyä ja liian pitkäsanaista. Pitkää tekstiä käyttäessä tutkija saattaa helposti päätyä ennen aikaisiin, vajaisiin ja perusteettomiin päätöksiin, sillä ihmisiä ei ole tehty prosessoimaan suurta määrää tietoa. Pitkä teksti kuormittaa ihmisen tiedon prosessointikykyä ja ajaa ihmisen etsimään yksinkertaisempia tiedon esitystapoja. Tutkija voi jopa ohittaa turhiksi luul-

tujen sivuja, jolloin on vaarana että tutkimukselle kriittinen tieto saatetaan jättää huomiotta. (Miles ym. 2014, 13.)

Hyvä datan esitysmuoto on kvalitatiivisen analyysin vankka pohja. Tiivistetty data voidaan esittää matriiseina, graafeina, kuvaajina ja verkkoina. Esitysmuodot kokoavat jäsennellyn tiedon heti saatavilla olevaan ja ymmärrettävään tiiviiseen muotoon. Esitysmuotojen avulla luki- ja voi luoda perusteltavissa olevia johtopäätöksiä. (Miles ym. 2014, 13.) Data voidaan myös esittää tiivistettynä tekstinä jos dataa ei esimerkiksi ole kvantifioitu. Tässä opinnäytetyössä on datan esitysmuotoina käytetty kuvaajia sekä tiivistettyä tekstiä. Esitysmuodot on valittu siksi, että ne tukevat toisiaan. Kuvaajien on tarkoitus havainnollistaa kvantitatiivista, eli määrämuotoista dataa. Tiivistetyn tekstin tarkoituksena on tukea kuvaajia selittäen kuvattuja ilmiöitä, sekä esittää kvalitatiivinen analysoitu data opinnäytetyössä.

Kuten datan tiivistys, datan esittäminen ja havainnollistaminen ovat osa analysointia. Analysoinnin aikana sarakkeiden ja rivien määrittäminen sekä datan esitys matriisissa ovat analyyttisiä päätöksiä jotka vaikuttavat kerättävän datan muotoon. Tutkimuksen alusta lähtien kvalitatiivisen datan tutkija tulkitsee mitä asiat tarkoittavat huomioimalla toistuvia sekä satunnaisia tapahtumia, ehdotelmia ja selityksiä. Tutkija huomioi nämä, mutta samalla on avoin ja skeptinen. Johtopäätökset ovat koko ajan saatavilla datasta. Aluksi epämääräisinä, implisiittisinä, mutta tutkimuksen edetessä enemmän määrin selvinä, eksplisiittisinä, tietoina. Saatavuudesta huolimatta johtopäätökset eivät välttämättä ole nähtävissä vasta kuin datan keräys on ohi, riippuen kerätyn datan koosta, keruumenetelmistä, tutkijan asiaan vihkiytymisestä ja asetetuista määräajoista. (Miles ym. 2014, 13.)

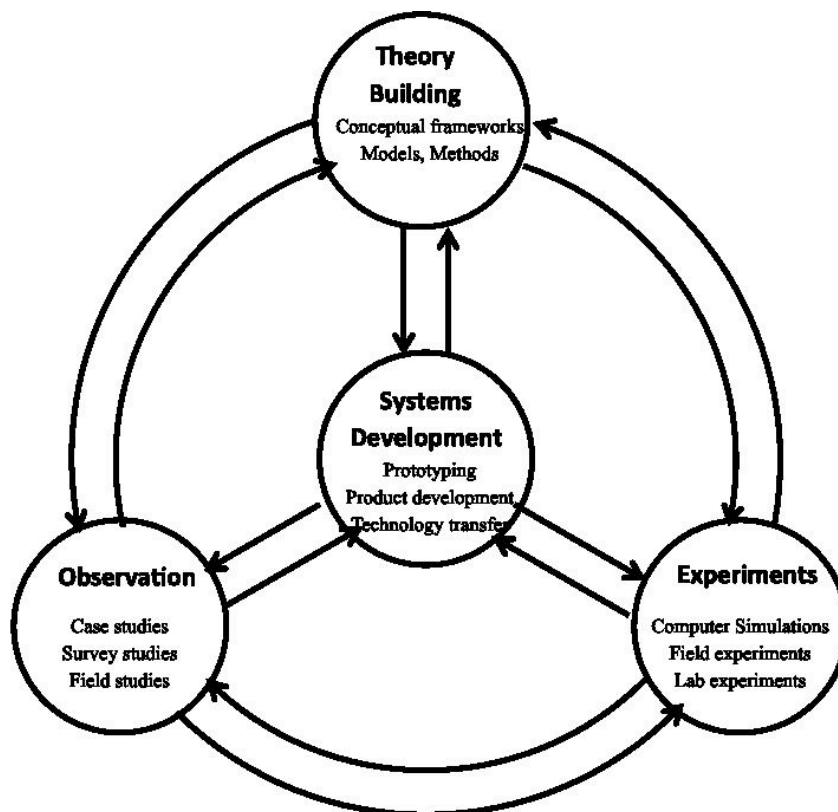
Johtopäätösten tuottaminen on vain osa analysointiprosessia. Johtopäätösten oikeellisuus on myös varmistettava. Varmistus johtopäätöksen oikeellisuudesta voidaan tehdä tutkijan muistijäljestä, tehdyistä muistiinpanoista tai todistamalla johtopäätökset kerätyn tai ulkopuolisen yksityiskohtaisen datan avulla. Oikeellisuuden avulla varmistetaan johtopäätösten validiteetti, muutoin johtopäätökset ovat vain käyttökeltottomia tarinoita siitä miten kävi. (Miles ym. 2014, 13 - 14.) Monet aloittelevat tutkijat kokevat johtopäätöksiensä uskottavuusongelmia. Uskottavuusongelmia voi ratkoa pohtimalla kilpailevia johtopäätöksiä induktiivisesti ja loogisesti. Induktiivisessa mallissa kerätty data jäsennetään eri tavalla ja katsotaan päädytäänkö samoihin johtopäätöksiin. Loogisella tavalla yritetään mieltä muita mahdollisia loogisia johtopäätöksiä ja kokeillaan tukeeko kerätty data aiemmin toteutetuilla analysointimenetelmillä näitä loogisia johtopäätöksiä. Jos vaihtoehtoiset mallit eivät tuota tulosta ja tukevat alkuperäisiä johtopäätöksiä, tutkimuksen validiteetti ja laatu on siten parempi. (Patton 1999, 1191.)

Kuvan 7 mukaisesti johtopäätösten teko tarvitsee pohjaksi kerättyä dataa, tiivistettyä dataa ja jäsennettyä havainnollistettua dataa, jotta johtopäätösten validiteetti voidaan taata. Opinnäytetyön laatu ja validiteetti taataan vertailemalla opinnäytetyön tuloksia haastatteluiden sekä havainnoinnin tuottamaan tietoon. Opinnäytetyön tuloksena syntyneen kehitysehdotuksen tulisi pystyä vastaamaan haastattelu ja havainnointi tutkimusmenetelmin kerättyä tiedosta johdettuihin käyttötapauksiin. Johdetut käyttötapaukset edustavat aitoja tapahtumia, joita palvelinvalvontasovelluksen käytössä esiintyy. Opinnäytetyön validiteetti todistetaan Keskustelu -luvussa.

4.5 Tutkimuksen kulku

Tämä opinnäytetyö on kvalitatiivinen tutkimus, jonka tavoitteena on vastata esitettyihin tutkimuskysymyksiin. Vastaukset tutkimuskysymyksiin saadaan keräten tutkimusmenetelmin dataa, jota analysoidaan vuorovaikutteisen datan analysointiotteen avulla sekä lopuksi tulkitaan. Opinnäytetyön teon ohella syntyy syvempi ymmärrys tutkittavaan aihealueeseen, tutkimusmenetelmiin.

Tämä opinnäytetyö toteuttaa kirjassa Systems Development in Information Systems Research (Nunamaker, Chen & Purdin 1991) esitettyä usean tutkimusmenetelmän kehikkoa tietojärjestelmätutkimuksessa. Kehikko yhdistää teoriataustan muodostamisen (theory building), testauksen (experiments), tutkimuksen (observation) ja järjestelmäkehityksen (systems development). Kehikon vuorovaikutteisuutta on havainnollistettu kuvassa 8.



Kuva 8: Usean tutkimusmenetelmän kehikko tietojärjestelmätutkimuksessa (Nunamaker ym. 1991, 94).

Opinnäytetyössä käytetty usean tutkimusmenetelmän kehikko tietojärjestelmätutkimuksessa koostuu neljästä eri osa-alueesta. Ensimmäinen osa-alue on teoriataustan muodostaminen (theory building). Teoriataustan luonnin aikana muodostuu myös uusia ideoita, konsepteja, menetelmiä ja malleja. Teoriatausta koostuu tiedosta, joka on yleistä tutkittavan aiheen ympärillä. Tästä syystä syvällisempi tieto tutkittavasta kohteesta voi olla rajallista.

Toinen osa-alue on testaus (experiments). Testatessa suoritetaan laboratorio- ja kenttäkokeita sekä luodaan simulaatioita tietokoneen avulla. Testaus sijoittuu teoriataustan ja tutkimuksen väliin. Testauksen avulla voidaan varmistaa teoriataustan avulla syntyneiden teorioiden paikkaansapitävyyttä, sekä testata tutkimuksen avulla syntyneitä tuloksia. Testauksen tuloksien avulla voidaan parantaa teorioita sekä testattavia järjestelmiä.

Kolmas osa-alue on tutkimus (observation). Tutkimus koostuu tutkimusmenetelmistä, kuten esimerkiksi tapaustutkimuksista, kenttätutkimuksista ja huomaamattomasta havainnoinnista. Havainnointia käytetään yleensä silloin, kun tutkittavasta kohteesta tiedetään vähän. Koska tutkimusmenetelmien mukaiset tutkimusasetelmat ovat luonnolliset, eli asioita tutkitaan niiden luonnollisessa ympäristössä, voidaan tuloksia pitää merkityksellisempinä.

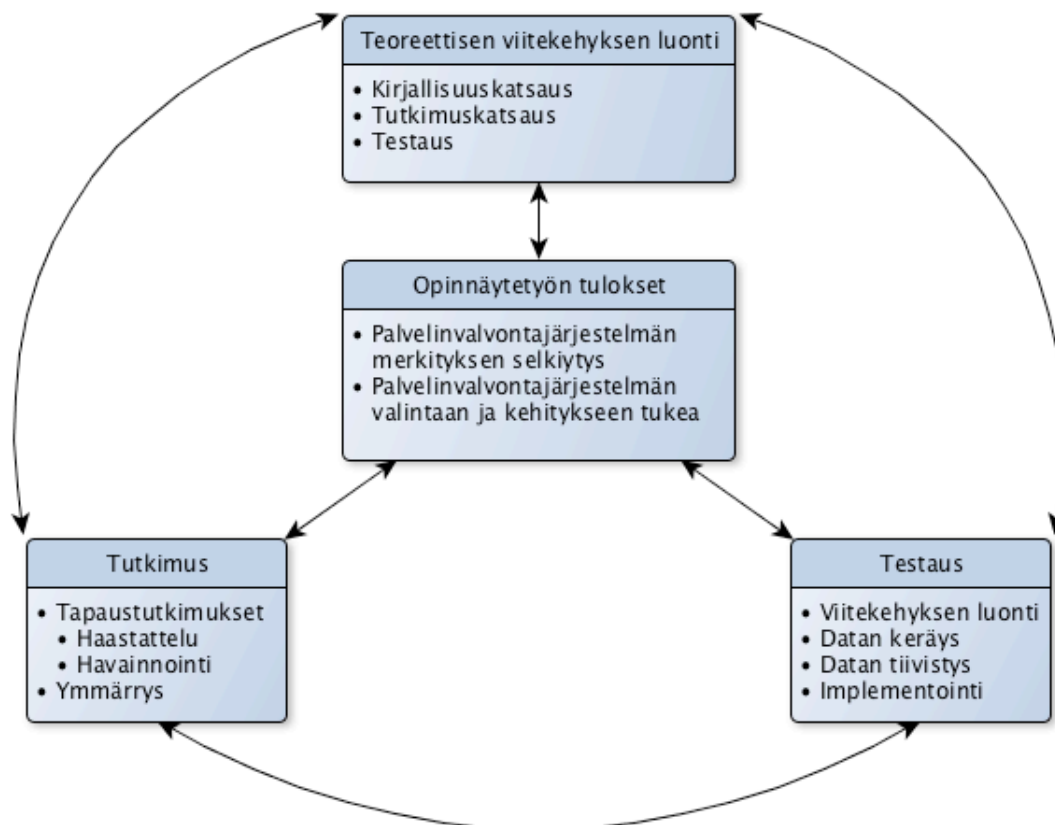
Neljäs ja viimeinen vaihe kehikossa on järjestelmäkehitys (systems development). Järjestelmäkehitys koostuu viidestä vaiheesta, jotka ovat konseptisuunnittelu, teknisen arkkitehtuurin toteutus, prototyypin luonti, tuotekehitys ja teknologiansiirto. Konseptisuunnittelussa yhdistetään ja mukautetaan teoria teknologian kanssa. Arkkitehtuurin rakennuksessa toteutetaan teknologinen vaatimus suunnitelmalle. Prototyypin tehtävänä on osoittaa, että konsepti on toteutettavissa teknologian avulla. Teknologiansiirrolla tarkoitetaan toimivan konseptin viemistä käyttöön esimerkiksi asiakkaille. (Nunamaker ym. 1991, 94 - 96.)

Opinnäytetyö vastaa esitettyihin tutkimuskysymyksiin

1. TK1: miksi palvelinvalvonta on tärkeä osa järjestelmänvalvontaa ja liiketoimintaa?
2. TK2: miten palvelinvalvontaa voidaan kehittää?

Opinnäytetyö tehdään usean tutkimusmenetelmän kehikolla, joka on esitetty kirjassa Systems Development in Information Systems Research (Nunamaker ym. 1991). Nunamakerin ym. Kehikosta poiketen opinnäytetyöhön ei kuulu kehikon viimeinen osa, joka on järjestelmäkehitys. Opinnäytetyön tulokset ovat hyödynnettävissä nykyisen palvelinvalvontajärjestelmän järjestelmäkehityksessä tai uuden palvelinvalvontajärjestelmän valinnassa. Nunamakerin ym. Ke-

hikkoa soveltaen, opinnäytetyön usean tutkimusmenetelmän kehikko tietojärjestelmätutkimuksessa toteutuu kuvan 9 mukaisesti.



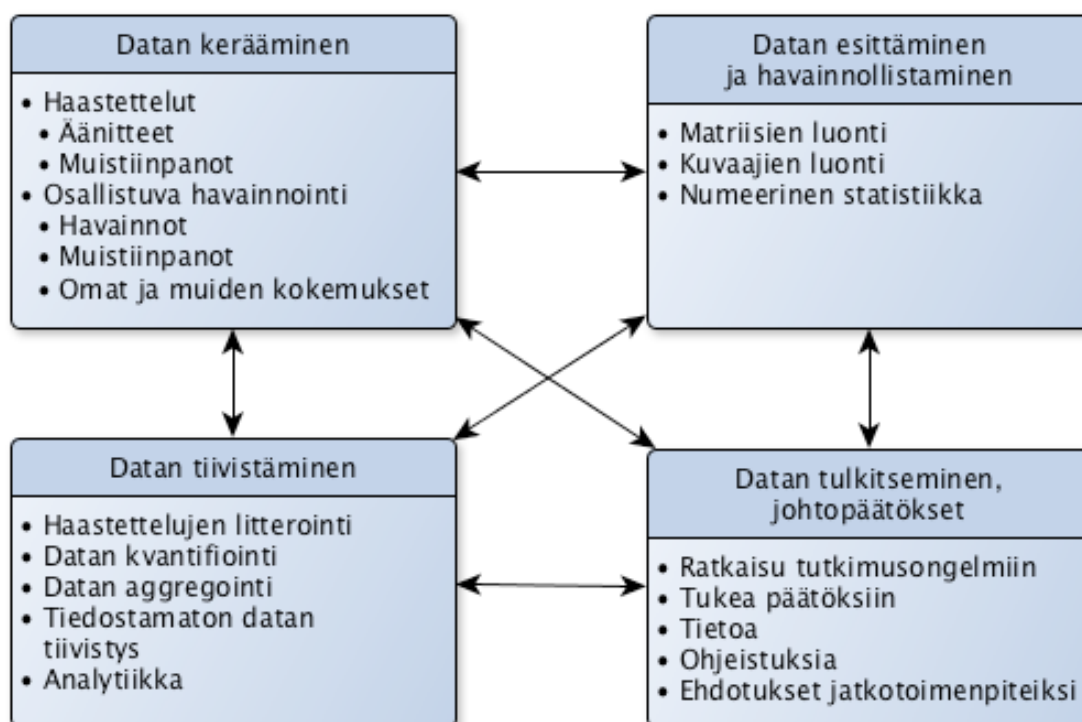
Kuva 9: Usean tutkimusmenetelmän kehikko opinnäytetyössä (kuva johdettu Nunamaker ym. 1991, 94).

Teoreettisen viitekehyksen luonti antaa perustiedot tutkittavaan asiaan. Se luo uusia ideoita sekä konsepteja, menetelmiä ja malleja kuten matemaattiset kaavat ja datamallit. (Nunamaker ym. 1991, 94 - 95.) Opinnäytetyössä teoreettinen viitekehys on muodostettu kirjallisuuskatsauksen avulla, aiheesta jo tehtyjen tutkimusten avulla sekä palvelinvalvontajärjestelmiä testaten. Teoreettisen viitekehyksen laajuutta rajaa aika, joka on opinnäytetyön aikataulun mukaisesti rajallinen.

Testaus on tutkimukselle tärkeä vaihe, sen avulla saadaan käytännön kokemusta ja tietämystä tutkittavasta aiheesta. Testaus tuo tietoa muihin tutkimusvaiheisiin, eritoten teorian muodostamiseen. Tietojärjestelmätutkimuksessa testaus usein tarkoittaa tutkittavan järjestelmän ja vaihtoehtoisten järjestelmien käyttöönottoa ja kokeilua käytännössä. (Nunamaker ym. 1991, 95.) Opinnäytetyössä järjestelmäkehitystä ei toteuteta, joten testauksen rooliksi on jäänyt luoda tietoa teoreettisen viitekehyksen tueksi sekä datan keräykseksi ja tiivistykseksi, jota hyödynnetään tutkimuksen datan analyysivaiheessa.

Tutkimuksen avulla voidaan tuottaa tietoa silloin, kun tutkittavasta asiasta ei ole tarpeeksi teoretietoa saatavilla. Tutkimus on järjestelmällistä, tieteellistä toimintaa. Käytännössä tutkimuksella tarkoitetaan esimerkiksi tapaustutkimusten, kenttätutkimusten ja kyselyjen tekoa valitulle kohderyhmälle. (Nunamaker ym. 1991, 95.) Koska opinnäytetyössä tutkitaan muun muassa itse kehitettyä sovellusta ja yrityksen yksilöityjä tarpeita, korostui tutkimusvaiheen tärkeys opinnäytetyön tulosten validiteetin ja reliabiliteetin takaamiseksi. Aiemman tutkimustiedon puuttuessa, on tapaustutkimusten teko ollut opinnäytetyölle ehdoton edellytys. Tapaustutkimuksia tehtiin kaksi kappaletta. Ensimmäinen tapaustutkimus tutkii omana tuottajana kehitetyn valvontasovelluksen käyttöönottoa yrityksessä RELEX Oy. Tapaustutkimus 1 on tehty vuonna 2013, jolloin nykyisin käytössä oleva palvelinvalvontajärjestelmä oli yrityksessä uusi. Aiemmin yrityksessä ei ollut käytössä palvelinvalvontajärjestelmää. Tapaustutkimuksen tavoitteena oli tutkia miten palvelinvalvonta muuttui järjestelmänvalvojan näkökulmasta ja minkälainen liiketoiminnallinen ja järjestelmänvalvonnallinen merkitys palvelinvalvontasovelluksen tuomisella oli yritykselle. Toinen tapaustutkimus tutkii kahdeksan palvelinvalvontasovelluksen ja nykyisten käytössä olevien palvelinvalvontasovellusten soveltuvuutta RELEX Oy:n tarpeisiin. Soveltuvuuden mittaus suoritetaan tekemällä vaatimusmäärittäminen palvelinvalvontasovellukselle sekä analysoimalla miten palvelinvalvontajärjestelmät täyttävät asetetut vaatimusmäärittäykset. Vaatimusmäärittäyksen toteutuminen on kuvattu liitteessä 3. Kummankin tapaustutkimuksen tulokset ovat luettavissa luvuista 4.5.1, Tapaustutkimus 1 ja 4.5.2, Tapaustutkimus 2.

Opinnäytetyön tutkimus alkaa teoreettisen viitekehyksen luonnilla. Sen tehtävänä on syventää tutkijan sekä lukijan tietoa tutkittavasta asiasta. Se myös luo uusia ideoita järjestelmäkehitystä varten. Teoreettisen viitekehyksen luomisessa auttoi myös testaus. Tutkimuksen datan keruu ja data-analyysi toteutettiin vuorovaikutteiden datan analysointiotteella. Analysointiotte on esitetty kirjassa *Qualitative Data Analysis* (Miles, Huberman & Saldaña 2014) sekä tämän opinnäytetyön luvussa 4.4, analysointi. Datan vuorovaikutteinen analysointiotte toteutuu opinnäytetyössä kuvan 10 mukaisesti.

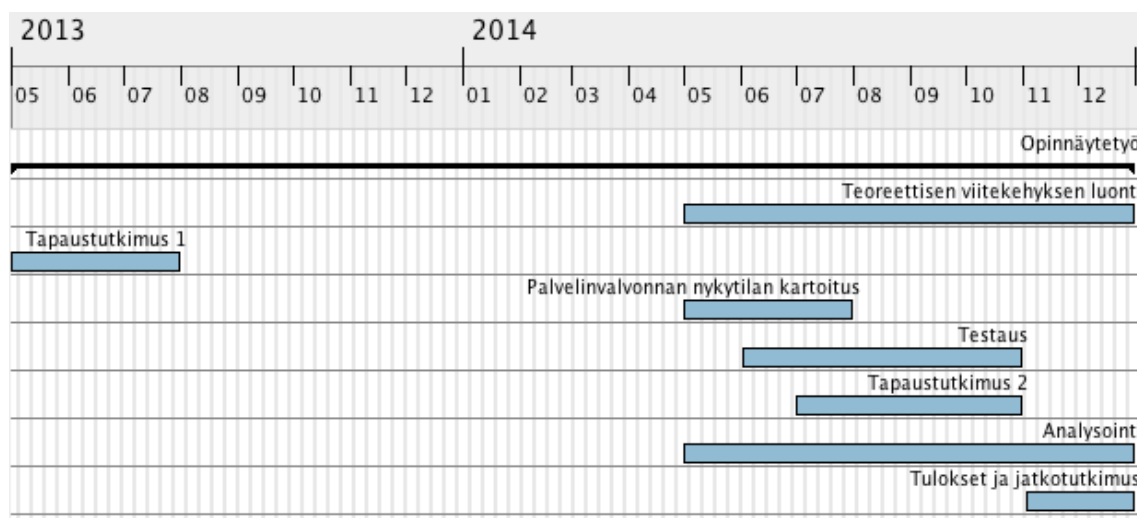


Kuva 10: Vuorovaikutteinen datan analysointiotte opinnäytetyössä (kuva johdettu Miles ym. 2014, 40).

Usean tutkimusmenetelmän kehikon viimeinen vaihe, järjestelmäkehitys yhdistää teknologisen ja teoreettisen tiedon luoden artefakteja kuten sovellukset ja prototyypit. (Nunamaker ym. 1991, 95.) Opinnäytetyössä ei toteuteta kehikon järjestelmäkehitysvaihetta. Opinnäytetyön tulokset ovat hyödynnettävissä uuden palvelinvalvontajärjestelmän valinnassa tai nykyisen palvelinvalvontajärjestelmän järjestelmäkehityksessä. Tämän lisäksi opinnäytetyön tulokset selkiyttävät palvelinvalvontajärjestelmän merkitystä yritykselle.

Opinnäytetyön teko ajoittuu ajalle 1.5.2014 - 31.12.2014. Kuvassa 11 on Gantt -kaavio, joka havainnollistaa opinnäytetyön vaiheiden ajoitusta. Selkeyden vuoksi kaaviossa ei näytetä vaiheiden alavaiheita. Tapaustutkimus 1 on tehty vuonna 2013, jolloin nykyisin käytössä oleva palvelinvalvontajärjestelmä oli yrityksessä uusi. Teoreettinen viitekehys tarkentuu koko opinnäytetyön ajan, suurin osa luvun työstä painottuu kuitenkin opinnäytetyön ensimmäiselle puoliskolle. Palvelinvalvonnan nykytilan kartoitusta työtetään opinnäytetyön alussa, sillä se auttaa hahmottamaan opinnäytetyön kokonaisuutta sekä tuo arvokasta tietoa opinnäytetyön muihin vaiheisiin. Testausta, eli palvelinsovelluksien sovellustestausta suoritetaan lähes koko opinnäytetyön ajan, sillä se tukee muita samanaikaisesti tehtäviä vaiheita. Vuorovaikutteisen analysointiotteen mukaisesti analysointia tehdään koko opinnäytetyön elinkaaren ajan, datankeruusta tuloksiin saakka. Tulosten sekä jatkotutkimuksen hahmotus alkaa opinnäytetyön

loppupuolella ja päättyy opinnäytetyön loppuun mennessä. Tulokset ja jatkotutkimusehdotukset tarvitsevat ovat riippuvaisia analysoinnin tuottamasta tiedosta. Tästä syystä se ajoittuu opinnäytetyön loppuun.



Kuva 11: Opinnäytetyön vaiheet ja ajoitus.

4.5.1 Tapaustutkimus 1

Ensimmäinen tapaustutkimus tutki omana tuotantona kehitetyn reaktiivisen ja proaktiivisen valvontasovelluksen käyttöönottoa yrityksessä RELEX Oy. Valvontasovelluksia käyttäviin henkilöihin kohdistuneiden haastattelujen avulla kävi selväksi, että palvelinvalvontasovelluksen käyttöönotto on onnistunut. Uuden palvelinvalvontasovelluksen tuominen organisaatioon on ollut helppoa, sillä vanhaa ja uutta palvelinvalvontamallia on voitu käyttää rinnakkain. Vanhan ja uuden palvelinvalvontasovelluksen suurimpana erona on ollut se, että vanha on perustunut täysin historiadatan avulla käsin tehtävään valvontaan. Se ei ollut reaktiivinen eikä proaktiivinen reaaliaikaiseen dataan, joten uusi valvontasovellus on automaationsa vuoksi helpottanut järjestelmänvalvojien työtä palvelinvalvonnassa. Vanhasta palvelinvalvontasovelluksesta ei ole uskallettu luopua siitä syystä, että se sisältää mittareita, joita uudessa palvelinvalvontasovelluksessa ei vielä ole toteutettu. Tämän lisäksi uudessa palvelinvalvontasovelluksessa ei ole käyttöliittymää, mistä kerättyä dataa voidaan tarkastella.

Päivittäiseen palvelinvalvontatoimintaan suurin muutos on tullut siitä, että näkyvyys palvelimien tiloihin on kasvanut. Uusi palvelinvalvontasovellus on reaktiivinen sekä proaktiivinen, joka tarkoittaa enemmän hälytyksiä palvelimilta. Osa hälytyksistä on ollut osoittautunut vääräksi, mutta on ymmärrettävä, että automaattinen valvonta vaatii jatkuvaa kehitystä jotta löydetään missä menee turhan ja merkityksellisen hälytyksen raja. Lisääntyneet hälytykset on kuitenkin koettu positiiviseksi asiaksi. Näiden asioiden johdosta palvelimien suorituskykyä ja

saatavuutta rajoittavien vakavien virhetilanteiden määrä on laskenut, joka hyödyttää niin järjestelmänvalvojia kuin asiakkaitakin. Reaktiivisuus antaa valmiudet reaaliaikaisen virheiden havaitsemiseen. Proaktiivisuus tukee virheiden ennaltaehkäisemistä asettamalla valvo- tuille asioille kynnsarvoja, joiden ylittyessä saadaan virhe ennen kuin valvottava palvelin on virhetilassa.

Nykyinen palvelinvalvonta sisältää järjestelmänvalvonnan perusmittarit. Usein on kuitenkin tullut tilanne jossa lisätietoja palvelimen tilasta tarvittiin, mutta sitä ei oltu kerätty, eikä siitä syystä saatavilla. Nykyinen palvelinvalvonta sai kiitosta luotettavuudestaan, se toimii hyvin siinä työssä mihin se on luotu. Omana tuotantona kehitetyn palvelinvalvontasovelluksen kehityksen on kuitenkin todettu vievän paljon työresursseja ja sen vuoksi todettiin, että on hyvä tutkia mitä olemassa olevat palvelinvalvontasovellukset tarjoavat ja olisiko niistä täyt- tämään RELEX Oy:n palvelinvalvontasovellukselle asetetut vaatimukset.

Haastattelujen ja havainnoinnin tulokset kertoivat, että palvelinvalvontasovelluksen uusien ominaisuuksien tarve on suuri. Tämä tarkoittaa, että nykyinen palvelinvalvontasovellus ei ole valmis ja se ei sellaisenaan palvele yrityksen tarpeita. Havainnoinnin avulla kerättiin myös kehitysehdotuksia liittyen palvelinvalvontasovelluksen käytettävyyteen.

Nykyisestä palvelinvalvontajärjestelmästä tehtiin seuraavia tärkeäksi koettuja yleisiä havain- toja tehtiin seuraavista asioista: PostgreSQL relaatiotietokantapalvelinsovellus aiheuttaa mää- rällisesti eniten hälytyksiä nykyisessä valvontajärjestelmässä. Valvontasovelluksen tuottama viestintä on hyvää, viestit ovat tarpeeksi informatiivisia järjestelmänvalvojille, jolloin niiden pohjalta on pystytty tekemään päätöksiä. Valvontasovelluksen konfigurointi on helppoa. Val- vontasovelluksen ohjelmistopäivityksen tekeminen on helppoa.

Jatkokehityshavaintoja tehtiin enemmän kuin yleisiä havaintoja. Tärkeitä jatkokehityshavain- toja olivat: kynnsarvojen tarkistus, sillä moni hälytys, kuten palvelimen korkean käyttöas- teen tai pitkäaikaisen PostgreSQL tietokantakyselyn hälytys eivät johtaneet toimenpiteisiin. Sähköpostitse tulevilla hälytysviesteillä tulisi olla linkki palvelinvalvontasovelluksen web- käyttöliittymän historia-kuvaajaan kyseisen mittarin osalta. Palvelinvalvontasovelluksen pal- velinkohtaisen konfiguraation tulisi olla muokattavissa web-käyttöliittymästä hallittavuuden helpottamiseksi. Palvelimesta olisi hyvä saada kuukausiraportti, josta käy ilmi resurssien käyt- töaste. Valvottavan kohdejärjestelmän kadotessa verkosta tulisi saada hälytys. Tuki laitetason valvonnalle tulisi olla mahdollista esimerkiksi Dell IDRAC integraation avustuksella. Tuotanto- asennuksissa kriittisistä valvottavista mittareista tulisi tekstiviestihälytysten olla saatavilla. Käyttöjärjestelmälokien tarkistusta tulisi suorittaa, jotta käyttöjärjestelmätason virheet voi- daan saada kiinni heti niiden ilmetessä.

Havaintojen lisäksi tutkimusmenetelmät tuottivat dataa, jonka avulla voidaan vastata tämän tapaustutkimuksen tutkimuskysymyksiin 1 ja 2.

TT1.TK1: miten omana tuotantona kehitetyn valvontasovelluksen käyttöönotto on vaikuttanut yrityksessä RELEX Oy päivittäiseen valvontatoimintaan? Päivittäisestä palvelinvalvontatoiminnasta on tullut reaaliaikaisuutensa ansiosta reaktiivista sekä proaktiivista. Palvelinvalvontasovelluksen avulla pystytään reaaliaikaisesti havaitsemaan ongelmat sekä ennakoimaan ongelmia ja siten ehkäisemään niitä. Reaaliaikaisen valvonnan myötä palvelimen ongelmat eivät enää kasaudu, jolloin palvelimen vakaa toiminta pystytään paremmin takaamaan. Ennen käsin tehtyyn palvelinvalvontaan käytetty aika vapauttaa työntekijöiden työaikaa muihin asioihin parantaen työntekijöiden tuottavuutta. Kerätyn historiadatan avulla voidaan tehdä kapasiteettisuunnittelua. Kapasiteettisuunnittelun avulla voidaan analysoida pystyykö palvelin vielä palvelemaan palvelinresurssien puitteissa siellä olevia asiakkaita tulevaisuudessa ja onko palvelimelle mahdollista tuoda vielä uusia asiakkuuksia. Tekstiviestihälytysten ansiosta järjestelmäviat huomataan myös tarvittaessa työaikojen ulkopuolella.

TT1.TK2: miten valvontasovellus hyödyttää RELEX Oy:n asiakkaita järjestelmänvalvojien näkökulmasta? Palvelinvalvontasovelluksen käyttöönoton jälkeen palvelimien sekä sovelluksien ennakoimattomat käyttökatkot ovat vähentyneet sekä RELEXin tarjoamien ohjelmistojen palvelutaso on kohentunut. RELEXin tarjoamat ohjelmistopalvelut ovat varmatoimisempia palvelinten ollessa valvottuina, sillä palvelinvalvontasovelluksen avulla viat voidaan havaita, määrittää ja selvittää nopeammin. Palvelinvalvontasovelluksen avulla palvelimien huoltoikkunat voidaan suunnitella siten, että ne vähiten häiritsevät asiakkaiden työskentelyä. Palvelinvalvontasovellus tarjoaa tärkeimmät mittarit, joilla voidaan todeta, että asiakas saa laadukasta ja toimivaa palvelua RELEXin tarjoamilta ohjelmistoilta. Mittareiden avulla pystytään myös osoittamaan, että palvelimet täyttävät ja ylittävät sovitut palvelutasoa koskevat sopimukset.

RELEX järjestelmien toimiessa odotetusti voidaan vähentää sekä asiakkaan että RELEXin tekemää työtä ja asiakas voi keskittyä liiketoimintaansa. Palvelimien toimiessa hyvin, RELEXin kehittämä sovellus toimii varmemmin ja asiakkaat ovat tyytyväisempiä. Näiden asioiden ollessa hyvin, myös yritys voi paremmin.

Kerätyn datan analysoinnin jälkeen voidaan todeta, että omatuotantona kehitetyn valvontasovelluksen käyttöönotto on onnistunut yrityksessä RELEX Oy. Palvelinvalvontasovelluksen käyttö on parantanut RELEXin palvelimien palvelutasoa. Palvelinvalvontasovellus kuitenkin vaatii lisäkehitystä, jotta se vastaisi paremmin yrityksen tarpeita. Kehitys on aikaa vievää ja täten on aiheellista tutkia täyttääkö jokin olemassa oleva palvelinvalvontasovellus yrityksen tarpeet.

4.5.2 Tapaustutkimus 2

Toinen tapaustutkimus tutkii yhdeksän palvelinvalvontasovellusten soveltuvuutta RELEX Oy:n tarpeisiin. Tarve tapaustutkimukselle lähti opinnäytetyössä ensimmäisen tapaustutkimuksen päädyttyä tulokseen, jossa palvelinvalvontasovelluksen kehitys on aikaa vievää ja täten tulisi tehdä vaatimusmäärittäminen sekä tutkia täyttääkö jokin olemassa oleva palvelinvalvontasovellus yrityksen määritetyt tarpeet. Tapaustutkimuksessa määritettiin RELEX Oy:n tarpeet valvontasovellukselle haastattelujen sekä havainnoinnin avulla. Tapaustutkimuksessa selvisi, että palvelinvalvontasovelluksissa on suuria eroja ja valvontasovellukset vastaavat valvontaominaisuuksiltaan usein eri tarpeisiin.

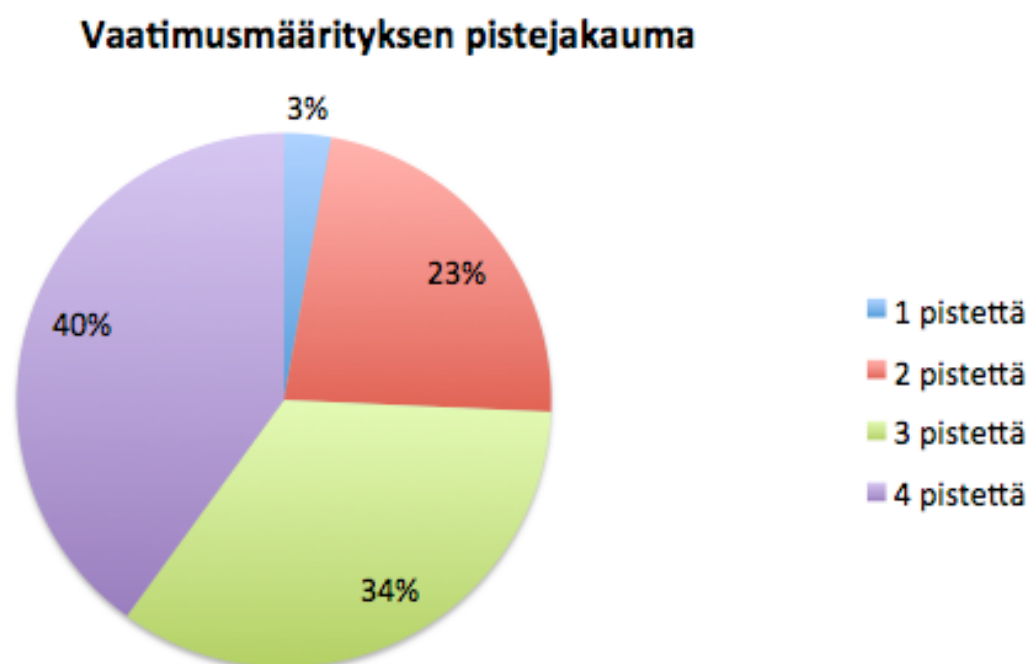
Palvelinvalvontasovellukset ovat usein kehitetty ratkaisemaan joitakin tiettyjä valvontatarpeita ja ovat rajoittuneita tai kykenemättömiä kun tarve on täysin erityyppiseen valvontatoimintaan. Palvelinvalvontasovelluksiin tehdyt laajennukset ovat usein tehty yhteisövoimin. Yhteisövoimin tehdyt laajennukset ovat olleet positiivinen asia, koska yhteisön avulla palvelinvalvontasovelluksiin on saatu paljon lisätoiminnallisuutta, jota ilman yhteisöä ei olisi. Yhteisövoimin tehtyjen laajennuksien kääntöpuolena on joissakin tapauksissa hidas kehitys, sekä että ne ovat puutteellisia. Laajennukset eivät aina vastaa varsinaisen palvelinvalvontasovelluksen vaatimuksia sen päivittyessä, rikkoen yhteensopivuuden ja aiheuttaen siten käyttökäytön kyseiseen laajennukseen. Jos laajennus on tehty avoimena lähdekoodin periaatteilla, voi yhteisö korjata ja kehittää laajennusta alkuperäisen kehittäjän puolesta. Kaikkien palvelinvalvontasovellusta käyttävien kaikkia tarpeita on vaikea miellyttää, jolloin joidenkin ominaisuuksien kohdalla on tehtävä kompromisseja. Siitä myös lähti tämä tapaustutkimuksen tarve. Tapaustutkimus ottaa selvää tarpeista, sekä miten eri ohjelmat toteuttavat näitä tarpeita. Lopuksi tehdään kompromissi sovellusten väliltä ja ehdotetaan valittavaksi sovellus, joka toteuttaa tarpeet parhaiten.

Palvelinvalvontasovelluksen vaatimusmäärittäminen tehtiin yhteistyössä päivittäin palvelinvalvontatoimintaa tekevien ihmisten kanssa. Vaatimusmäärittäminen saatiin keräämällä tietoa käyttäjiltä haastattelujen ja työtehtävien lomassa tehdyn havainnoinnin avulla. Vaatimusmäärittäminen tuloksena oli 35 vaatimusta, jotka palvelinvalvontasovelluksen tulisi täyttää. Ilmi käyneet tarpeet pisteytettiin tärkeysluokkiin, koska vaatimukset eivät olleet samanarvoisia. Pisteytys toteutui seuraavanlaisesti

Pistemäärä	Selite
1 piste	ei vaadita, ei käytetä tai lisätieto
2 pistettä	ei vaadita, mutta käytetään
3 pistettä	vaaditaan, olennainen komponentti
4 pistettä	vaaditaan ehdottomasti

Taulukko 2: Vaatimusmäärittelyn pisteytys.

Määrittelyt jakautuivat pistemääriin kuvan 12 mukaisesti. Eniten pisteitä, 40% vaatimuksista, saivat ominaisuudet, jotka palvelinvalvontasovelluksen tulisi ehdottomasti täyttää. Toiseksi eniten pisteitän, 34% vaatimuksista sai olennaisesti auttavat komponentit. Loput 26% vaatimuksista edustaa vähemmistöä. Tähän joukkoon kuuluvat vaatimukset, joita valvontasovellukselta ei ehdottomasti vaadita mutta käytetään jos sellainen on. Sekä lisäksi vaatimukset, jotka toimivat lisätietona vaatimusmäärittelymatriisissa, mutta eivät estä palvelinvalvontasovellusta toimimaan vaatimusten mukaisessa työssään.



Kuva 12: Vaatimusmäärittelyn pistejakauma.

Pisteytyksien keskiarvoista voitiin päätellä, että kategorioissa valvontasovelluksen käyttöliittymän ominaisuudet sekä käyttöjärjestelmän ja suorituskyvyn valvonnassa, koettiin keskimäärin olevan tärkeimmät vaatimukset tulevalle palvelinvalvontaratkaisulle. Palvelinvalvontasovelluksen vaatimusmäärittelysten pisteytysten keskiarvot kategorioittain olivat seuraavat

Vaatusmääritysten kategoriat	Pisteytyksen keskiarvo
Valvontasovelluksen ominaisuudet	2,7
Valvontasovelluksen käyttöliittymän ominaisuudet	3,5
Sovelluksien ja palvelujen valvonta	3,2
Käyttöjärjestelmän ja suorituskyvyn valvonta	3,4
Kaikkien vaatimusten keskiarvo	3,2

Taulukko 3: Vaatusmäärityskategorioiden pisteytysten keskiarvot.

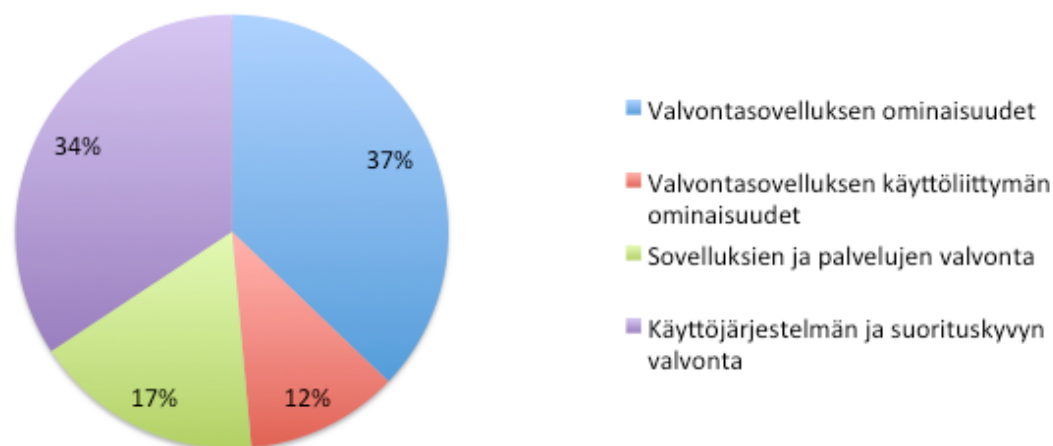
Vertailussa oli mukana kahdeksan yleisesti käytettyä sekä aktiivisesti kehityksessä olevaa palvelinvalvontasovellusta. Nämä kahdeksan valvontasovellusta valittiin testiryhmään sillä perusteella, että ne todennäköisesti toteuttavat dokumentaationsa mukaan RELEX Oy:n liitteessä 3 asettamia vaatimuksia vähintään kohtalaisesti. Tämän lisäksi vertailussa oli mukana RELEXin itse kehitetty palvelinvalvontasovellus, jotta voidaan määrittää nykyisen palvelinvalvonnan taso vaatimuksiin nähden. Karsintavaiheen ryhmä koostettiin Internetistä hakukoneilla löytyneillä osumilla, sekä seuraamalla alan uutisia ja foorumeita. Karsintavaiheen jälkeen testiryhmässä mukana olleet palvelinvalvontasovellukset olivat

#	Palvelinvalvontasovellus
1	Nagios
2	Munin
3	Monit + M/Monit
4	Zabbix
5	Zenoss
6	Osmius
7	Groundwork
8	Icinga
9	RELEXin kehittämä palvelinvalvontajärjestelmä

Taulukko 4: Tutkitut palvelinvalvontasovellukset.

Vaatusmäärityksen vaatimukset jaettiin neljään yläkategoriaan jotka käyvät ilmi kuvasta 13. Kuva osoittaa miten palvelinvalvontasovelluksen sisäiset ominaisuudet esittävät merkittävää 39% osaa vaatimuksista. Seuraavaksi eniten, 33% vaatimuksista lokeroitui käyttöjärjestelmän ja suorituskyvyn valvontaan liittyviin asioihin.

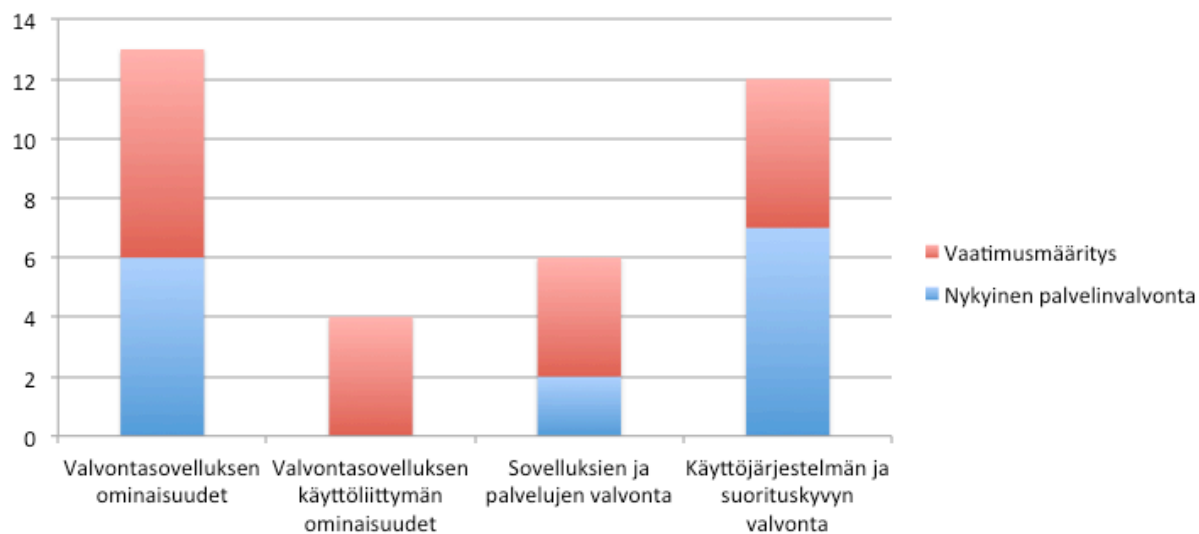
Vaatusmäärityksen kategorijakauma



Kuva 13: Vaatusmäärityksen kategorijakauma.

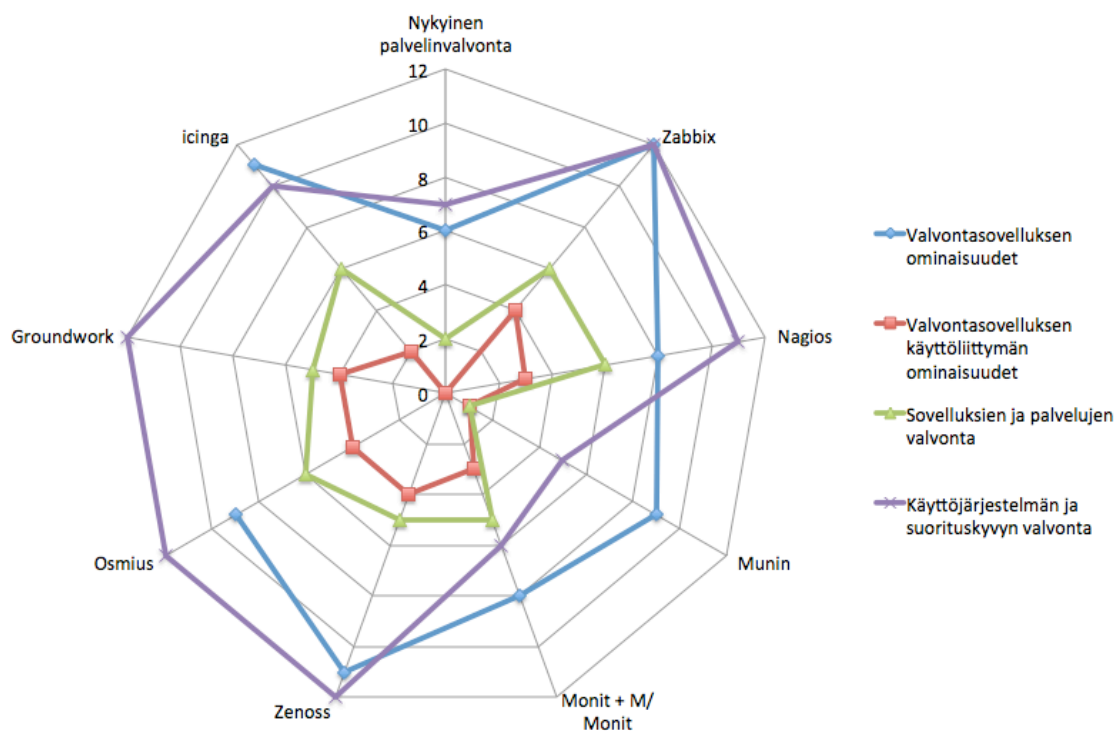
Valvontasovelluksen ominaisuudet -kategorialla tarkoitetaan valvontasovelluksen yleisiä asioita, kuten onko sovellus ilmainen, onko sovelluksella kattava dokumentaatio, voiko sovelluksella ennustaa mitattuja resursseja. Käyttöliittymän ominaisuudet asettavat käyttöliittymälle toiminnallisia vaatimuksia. Sovelluksien ja palvelujen valvonnalla kartoitetaan yritykselle tärkeiden sovelluksien, kuten tietokantapalveluiden, RELEXin oman sovelluksen valvontaa ja raportointia käsitteleviä vaatimuksia.

Nykyinen palvelinvalvonta kattaa vaatusmäärityksen huonosti, kuten kuvasta 14 voi todeta. Yhdenkään kategorian vaatimuksista ei pystytä täyttämään edes tyydyttävästi. Erityisesti mainittavan arvioista on, että käyttöliittymävaatimuksia ei pystytä täyttämään ollenkaan. Sovellusten sekä palvelujen valvonta on erityisen huonossa tilassa nykyisessä palvelinvalvonnassa.



Kuva 14: Nykyisen palvelinvalvonnan vastaus vaatimusmäärittäykseen.

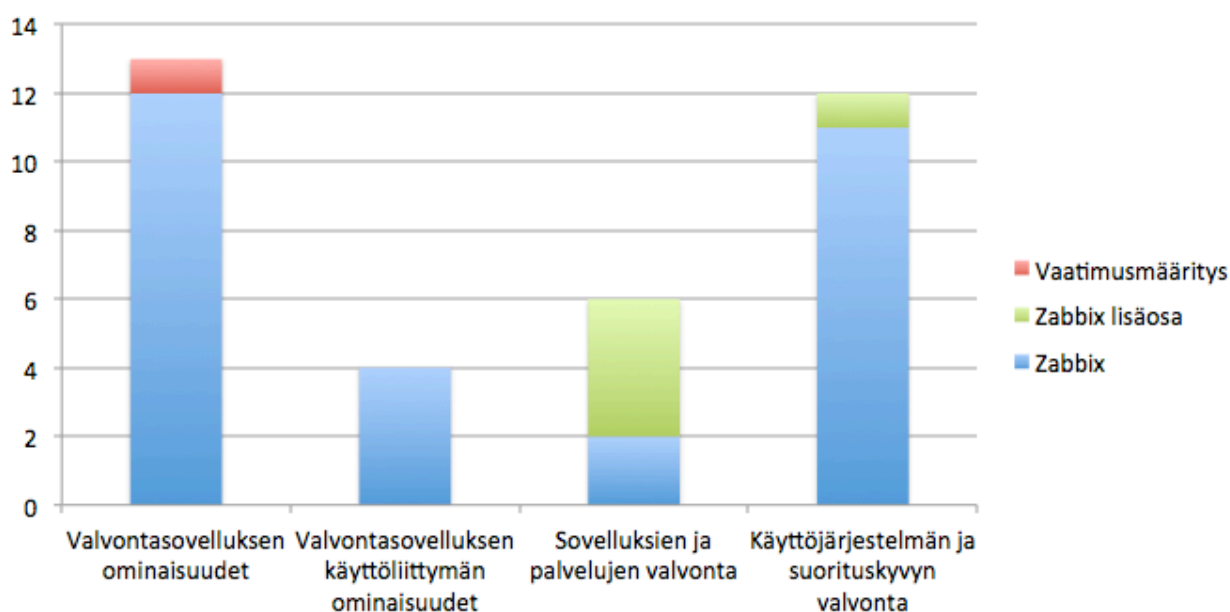
Vaatimukset asetettiin taulukkoon riveille ja palvelinvalvontasovellukset sijoitettiin sarakkeille. Tämän jälkeen tutkittiin jokaisen palvelinvalvontasovelluksen dokumentaatiota tai testattiin palvelinvalvontasovellusta, onko sovellus kykeneväinen täyttämään vaatimuksen vai ei. Myöskin merkittiin tieto, jos vaatimus voitiin täyttää käyttämällä lisäosaa valvontasovelluksessa. Yleisimmäksi puutteeksi nousi palvelinvalvontaohjelmistoissa ennustuksen puute. Kuvassa 15 on havainnointi miten eri ohjelmat ovat pystyneet toteuttamaan asetettuja tarpeita.



Kuva 15: vaatimusten toteutus palvelinvalvontasovelluksittain.

Edellä mainitut tiedot pohjautuvat liitteessä 3 olevaan taulukkoon, jossa riveille on asetettu tarpeet ja sarakkeisiin palvelinvalvontasovellukset. Tämän jälkeen on tehty tutkimus vaatimuksen toteutumisesta ja sijoitettu taulukkoon totuusarvo.

Tehdyn vaatimusmäärittelyn parhaiten taitava sovellus on Zabbix. Zabbix täyttää vaatimukset lähes täydellisesti. Zabbix on valvontasovelluskehikko, joka pystyy vastaamaan monenlaiseseen tarpeeseen. Zabbix on erityisen muuntautumiskykyinen. Samasta syystä myös valvontasovelluksen käytön oppiminen vie aikaa. Zabbixin ymmärtämiseen ja opetteluun kuluu aikaa, mutta perusasioiden ymmärryksen jälkeen on mahdollista tehdä varsin monimuotoista palvelinvalvontaa. Zabbix itsessään sisältää valmiita valvontatyökaluja erilaisten palvelimien, käyttöjärjestelmien sekä verkon aktiivilaitteiden valvontaan. Tämän lisäksi Zabbixin ympärille on muodostunut yhteisö, joka ylläpitää Zabbixiin kehitettyjä lisäosia tai liitännäisiä. Kuvasta 16 voidaan todeta miten Zabbix tai sen lisäosa toteuttaa määritetyt vaatimukset.



Kuva 16: Zabbix valvontasovelluskehikon vastaus vaatimusmäärittelyyn.

Tehdyn tutkimuksen avulla voidaan vastata tapaustutkimuksen tutkimuskysymyksiin.

TT2.TK1: miten nykyinen palvelinvalvonta vastaa yrityksen tarpeisiin?

Yrityksen nykyinen palvelinvalvonta vastaa yrityksen tarpeisiin heikosti. Tehdyn analyysin tuloksista voidaan päätellä, että nykyinen valvonta kattaa alle 50% ominaisuuksista, joita palvelinvalvontasovelluksen tulisi ehdottomasti kattaa. Käyttäjät kokevat nykyisen valvonnan toimivan hyvin niiden asioiden osalta, jotka on toteutettu, mutta uusien valvontakohteiden lisääminen on hidasta ja erittäin työlästä. Vakavimpia puutteita olivat seuraavat asiat. Kunnol-

lisen käyttöliittymän puuttuminen. Käyttöliittymästä tulisi voida nähdä yhdellä näkymällä koko IT-infrastruktuurin tila, missä on ongelmia ja minkälaisia ongelmat ovat luonteeltaan. Käyttöliittymän avulla tulisi pystyä lisäämään valvottavia kohteita sekä asettamaan hälytysrajoja kohteille. Tapahtumien ennakointia ja ennustamista ei voinut tehdä historiadataa hyödyntäen, vaikka tarvittava data on jo olemassa. Lisäksi RELEX Oy:n kehittämää ja ydinliiketoimintaan kuuluvaa logistiikan toimialan sovellusta ei voinut valvoa.

RELEX Oy:n kehittämän sovelluksen asiakasympäristöjä haluttaisiin myös valvoa. Kyseessä on suljetun lähdekoodin kaupallinen sovellus, jonka vuoksi valmista ratkaisua tähän ei ole olemassa. Valvonnan tulisi olla helposti tai automaattisesti asiakasympäristöihin päälle kytkettävissä. Valvontasovellukseen käytettävä työmäärä ei saisi kasvaa suhteessa palvelinten tai RELEX asiakasympäristöjen määrään.

TT2.TK2: miten yrityksen palvelinvalvonta kehittyisi palvelinvalvontasovellusta vaihtamalla?

Tehty tutkimus osoittaa, että yksi sovellus, Zabbix, täyttää lähes kaikki asetetut vaatimukset. Zabbixin avulla palvelinvalvonnan raportointi ja konfigurointi saadaan yhteen käyttöliittymään. Zabbix kattaa kaikki käyttöliittymälle, käyttöjärjestelmän ja suorituskyvyn valvonnalle sekä sovelluksien ja palvelujen valvonnalle asetetut vaatimukset. Valvontasovelluksen ominaisuuksista Zabbix ei pysty vastaamaan ennustamisen tarpeeseen vaatimusmäärittelyn tarkoitamalla tavalla. Zabbixia ei ole kehitetty ennustamiseksi, joten kyseinen ominaisuus puuttuu valvontajärjestelmästä. Hälytysrajoihin on mahdollista asettaa laskukaava, jonka avulla saadaan hälytys kun ongelma on nähtävissä. Mutta koska kyseessä on järjestelmän sallima kiertotie, jossa ennusteen kehitystä ei voi seurata, ei vaatimusta ei täytetä.

Palvelinvalvonta monipuolistuisi huomattavasti ottamalla käyttöön uuden Zabbix palvelinvalvontasovelluksen. Zabbix tarjoaa käyttöliittymän valvonnan konfiguroimiseen sekä raportointiin. Zabbixin avulla on mahdollista valvoa mitä vain laitetta tai sovellusta, jonka tilaa voi periodisesti kysyä jollakin protokollalla. Zabbix tuo palvelinvalvontaan proaktiivisen, eli virheiden ennaltaehkäisevän otteen parantaen valvottavien järjestelmien toimintakykyä. Zabbixin raportoinnin avulla on mahdollista tehdä kapasiteettisuunnittelua, ennakoida yksittäisen laitteen tai koko IT-infrastruktuurin tasolla kapasiteetin riittävyyttä.

Zabbixin monipuolisuus tuo kääntöpuolena mukanaan myös korkean oppimiskynnyksen. Oppimiseen kuluu reilusti aikaa ja konfigurointi on aluksi hidasta ennen kuin ymmärtää järjestelmän perusteet. Aluksi uusien valvottavien mittareiden määrittely voi tuntua erityisen hankalalta ja hitaalta, mutta määrittelyn valmistuttua siihen ei yleensä enää tarvitse koskea ja se skaalautuu sellaisenaan kaikkiin palvelimiin tai sovelluskohtainen mittari kaikkiin ilmentymiin kyseisestä sovelluksesta. Erityisen hyvä ominaisuus Zabbixissa on automaattinen havainnoitsi-

ja. Sen avulla voi luoda dynaamisia seurattavia kohteita sekä kuvaajia esimerkiksi palvelimisista, sovelluksista tai muista valvottavista asioista. Tämä parhaimmillaan syrjäyttää valvontakohdekohtaisen konfiguraation tarpeen täysin, säästäten aikaa.

5 Tulokset

Opinnäytetyön tarkoituksena on ollut kehittää palvelinvalvontatoimintaa yrityksessä RELEX Oy. Nykyinen palvelinvalvonta ja palvelinvalvontasovellus on koettu riittämättömäksi niin toiminnallisesti kuin valvottavien asioiden osalta. Opinnäytetyön tutkimuskysymykset on johdettu aidon työelämän ongelman pohjalta. Opinnäytetyössä tehty tutkimus ja sen tuottamat tulokset pyrkivät ratkaisemaan tämän ongelman. Opinnäytetyössä esitettyjen tutkimuskysymysten vastaukset on johdettu opinnäytetyössä tehtyjen tapaustutkimusten tulosten pohjalta. Opinnäytetyön tutkimuksen luonteen vuoksi tapaustutkimus koettiin hyväksi tutkimusmenetelmäksi tiedon saantiin sekä luontiin. Tapaustutkimuksessa käytettiin empiirisinä tutkimusmenetelminä haastattelua, havainnointia sekä tämän lisäksi myös teoreettisen luonteensa vuoksi viitataan paljon kirjallisuuteen.

TK1: miksi palvelinvalvonta on tärkeä osa järjestelmänvalvontaa ja liiketoimintaa. Ensimmäinen tutkimuskysymys selkeyttää palvelinvalvonnan merkitystä järjestelmänvalvonnallisesta sekä liiketoiminnallisesta näkökulmasta. Tutkimuksen tulokset auttavat järjestelmänvalvojia, päätöksentekijöitä sekä muita sidosryhmiä ymmärtämään palvelinvalvonnan merkityksen organisaatiossa. Vastaus tähän tutkimuskysymykseen on saatu opinnäytetyöhön tehdyn tapaustutkimus 1:n avulla. Ensimmäisen tapaustutkimuksen tulokset olivat seuraavat:

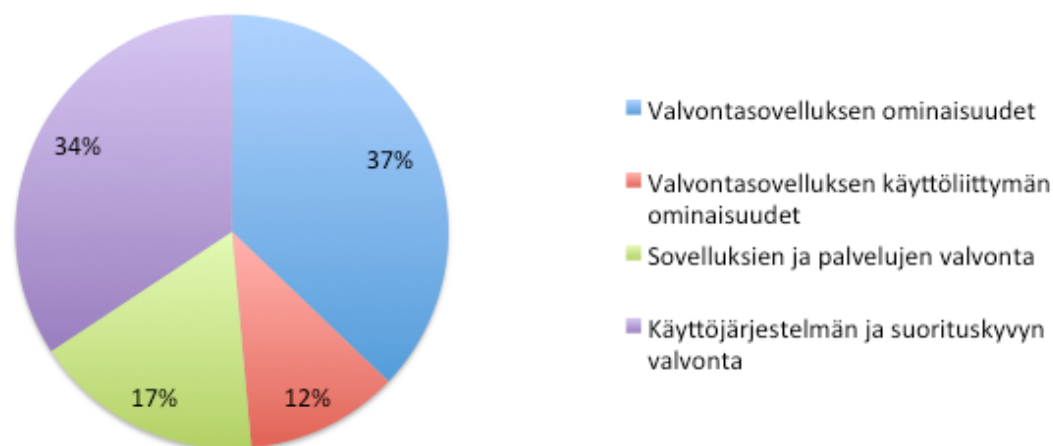
Toimiva palvelinvalvonta takaa järjestelmien sujuvan toiminnan. Virhetilanteista saadaan reaaliaikaisesti tieto, jolloin ongelmaa voidaan alkaa ratkaisemaan heti kun siitä on saatu tieto. Reaaliaikaisen havaitsemisen vuoksi ongelmat eivät kasaudu ja näy epämääräisenä virhetilana, vaan yhden ongelman ilmenemisen syy on helpompi löytää ja siten ehkäistä. Hyvä palvelinvalvonta tähtää myös ehkäisemään ongelmien synnyn, siksi valvontasovelluksen tulisi mitata kaikkea suorituskäytön sekä järjestelmän tarjoamiin palveluihin liittyviä asioita. Mitatuille asioille tulisi asettaa hälytysrajat siten, että potentiaalisista virhetilanteista saadaan tieto ajoissa ennen kuin ongelma on muodostunut. Mutta kuitenkin niin, että vääriä hälytyksiä ei tulisi liikaa. Järjestelmäresurssien, kuten kiintolevyn täyttöastetta tulisi myös ennustaa. Jos datasta on nähtävissä sopivin ennustemallein, että kiintolevytila on loppumassa asetetun ajanjakson sisällä, siitä tulisi saada hälytys. Jos valvottavana on jonkun muun, esimerkiksi asiakkaan oma laitteisto, voidaan palvelinvalvontasovelluksen avulla ongelmatilanteissa osoittaa ammattitaitoisesti missä ongelma piilee ja ehdottaa selkeitä ongelmanratkaisutapoja. Hyvä palvelinvalvonta näkyy asiakkaalle päin toimivana palveluna tai järjestelmänä.

Liiketoiminnan näkökulmasta hyvä palvelinvalvonta tarkoittaa, että asiakkaille tarjotuissa palveluissa on vähemmän käyttökatkoksia, eli palvelu on saumattomampaa. Virhetilanteen sattuessa ongelmanratkaisunopeus sekä -kyky kasvavat. Valvontasovelluksen hoitaessa palvelinvalvontaa, tarvitaan vähemmän henkilöstöresursseja valvomaan palvelimien tilaa sekä ratkaisemaan ilmeneviä ongelmia. Palvelinvalvontasovellus myös vakioi valvonnan, jolloin palvelimen tilanteen arviointi ei jää yksilön harkinnan varaan. Palvelinvalvonta saa valvontasovelluksen johdosta monia uusia piirteitä. Palvelinvalvonta on johdonmukaista, eli havainnoitavista asioista saadaan määritetyt virhetilanteet aina kiinni. Palvelinvalvonta on selkeätä, eli virheviestien syy ja tiedotustapa on yhdenmukaisesti raportoitu. Palvelinvalvonta on asiallista sekä asiantuntevaa, eli virheviestit on vakioitu sisältämään tarpeellisen tiedon ja virheviestin totuusperään voidaan luottaa. Palvelinvalvonta on kriittistä, eli tarvittaessa tarkoin mietittyä sekä tieteellisen tarkkaa. Palvelinvalvonta on tehokasta, eli resursseja ei käytetä liikaa, vaan huolellisesti. Edellä mainituista syistä voidaan todeta, että palvelinvalvonta on taloudellista, sen avulla liiketoiminta voi paremmin.

TK2: miten palvelinvalvontaa voidaan kehittää. Toinen tutkimuskysymys tuottaa vaatimusmäärittelyn palvelinvalvontasovellukselle ja löytää sopivimman palvelinvalvontasovelluksen vastaamaan vaatimusmäärittelyyn. Vastaus tähän tutkimuskysymykseen on saatu opinnäytetyöhön tehdyn tapaustutkimus 2:n avulla. Toisen tapaustutkimuksen tulokset olivat seuraavat:

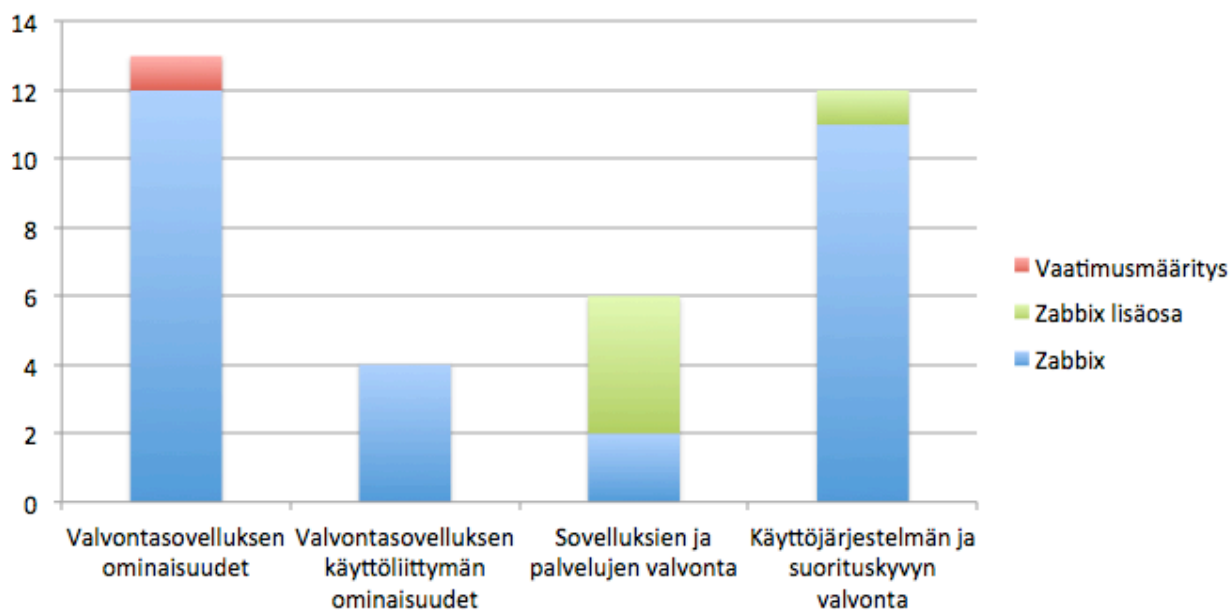
Vaatimusmäärittely tuotti kaiken kaikkiaan 35 kappaletta vaatimuksia. Vaatimukset kategorisoi-
ttiin neljään eri kategoriaan, jotka olivat valvontasovelluksen ominaisuudet, valvontasovelluk-
sen käyttöliittymän ominaisuudet, sovelluksien ja palvelujen valvonta sekä käyttöjärjestel-
män ja suorituskyvyn valvonta. Vaatimuksien kategoriajakaumaa on havainnollistettu kuvassa
17.

Vaatimusmäärittelyn kategorijakauma



Kuva 17: Vaatimusmäärittelyn kategorijakauma.

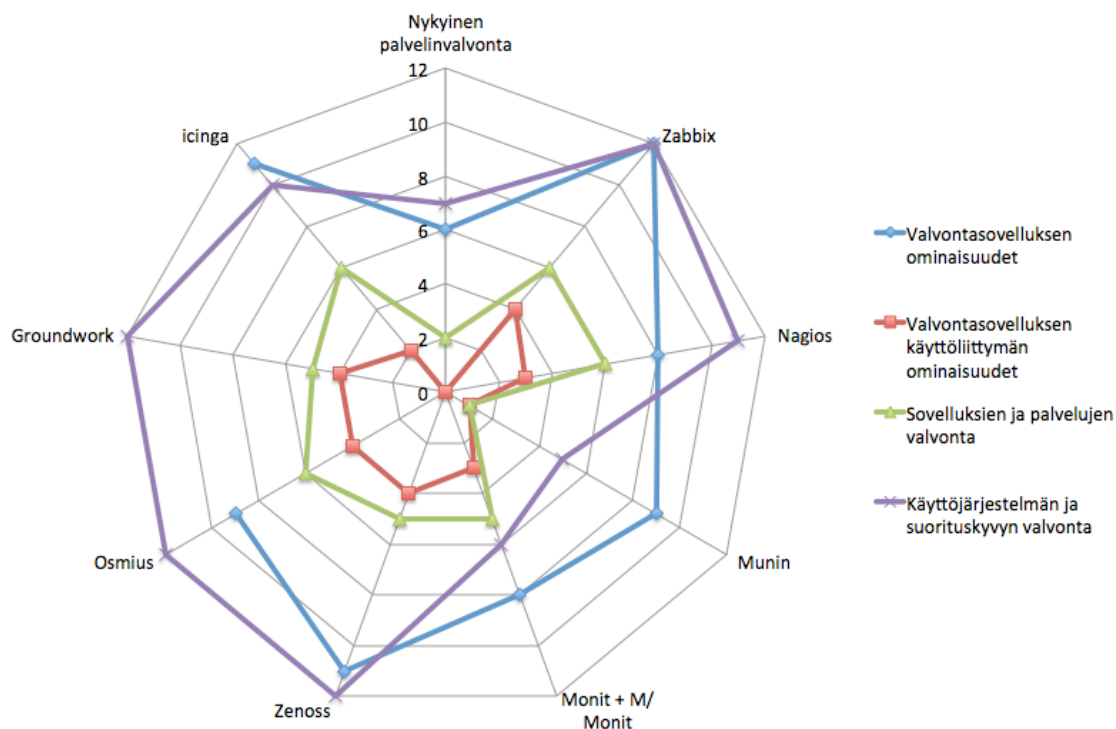
Tutkimuksessa vertailussa olleet sovellukset pystyivät vaihtelevasti toteuttamaan vaatimuksia. Zabbix valvontasovellus oli kuitenkin ylitse muiden. Kuvassa 18 on esitetty vaatimusmäärittelyn vaatimukset kategorioittain, sekä miten Zabbix on pystynyt vastaamaan asetettuihin vaatimuksiin. Suurimman osan vaatimuksista Zabbix täytti perustoiminnallisuudellaan ja loput lisäosien avulla. Zabbix ei kuitenkaan pystynyt täyttämään yhtä vaatimusta, joka oli ennustamisen tarve valvontasovelluksen ominaisuudet -kategoriasta. Valvontasovelluksen salliman kiertotien avulla hälytysrajoihin on mahdollista asettaa laskukaava, jonka avulla saadaan hälytys kun ongelma on nähtävissä, eli ennustettavissa. Valvontasovelluksen ei voida kuitenkaan todeta tukevan ennustamista koska ennusteen kehitystä ei voi seurata ja täten sitä ei voi hyödyntää resurssisuunnittelussa. Koska Zabbix käyttää datan taltiointiin PostgreSQL relaatio-tietokantaa, historiadata on tietokannasta saatavissa ja hyödynnettävissä myös valvontasovelluksen ulkopuolella. Esimerkiksi ennustamista voidaan tehdä erillisen sovelluksen avulla.



Kuva 18: Zabbix-valvontakehikon vastaus vaatimusmäärittämiseen.

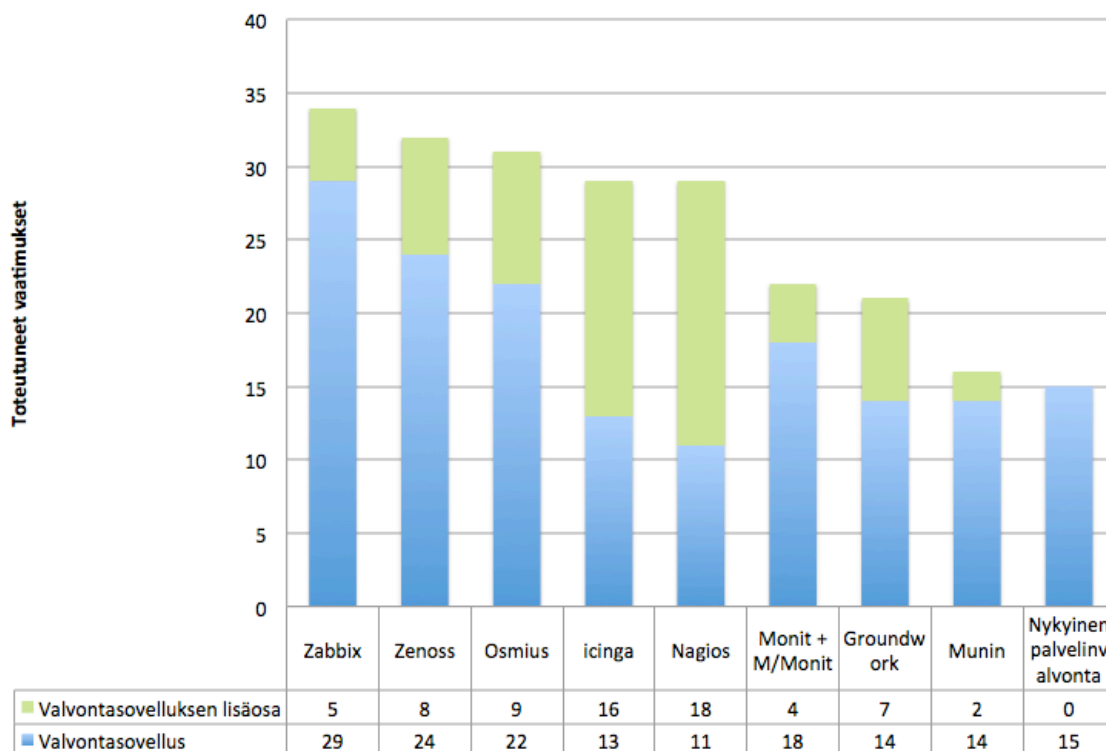
Zabbixin onnistuessa täyttämään vaatimukset erittäin hyvin, on suositeltavaa ottaa käyttöön Zabbix-valvontakehikko. Zabbixin palvelinvalvontaa voidaan suorittaa rinnakkain nykyisen palvelinvalvonnan kanssa, kunnes uuden palvelinvalvontasovelluksen käyttöönotto on suoritettu ja todetaan, että uusi palvelinvalvontajärjestelmä vastaa sille asetettuja vaatimuksia sekä odotuksia.

Muut palvelinvalvontasovellukset valvontajärjestelmät toteuttivat vaatimusmäärittäystä vaihtelevasti. Zenoss täytti vaatimuksia erittäin hyvin, mutta Zabbix osoittautui palvelemaan asetettuja vaatimuksia vieläkin paremmin. Zabbixin eduksi koitui myös, että suuri osa vaatimuksista täytettiin ilman lisäosien käyttöä. Kuvasta 19 näkee miten muut palvelinvalvontasovellukset ovat suoriutuneet vaatimusmäärittäyksestä kategorioittain.



Kuva 19: Valvontasovelluksien vaatimusmäärittelyn toteuma kategorioittain.

Valvontasovelluksien kokonaispisteitys järjestettynä parhaimmin vaatimuksen toteuttavasta huonoimpaan on havainnollistettu kuvassa 20. Pisteytyksessä tasapisteiden sattuessa on pidetty enemmän lisäosia käyttävää valvontasovellusta vähemmän sopivampana.



Kuva 20: Valvontasovelluksien vaatimusmäärittelyn toteuman kokonaispisteitys.

Lopuksi voidaan todeta, että opinnäytetyö onnistui tavoitteissa, jotka sille oli asetettu. Vastaukset esitettyihin tutkimuskysymyksiin saatiin käyttämällä onnistuneesti usean tutkimusmenetelmän kehikkoa. Usean tutkimusmenetelmän kehikon tutkimusprosessin mukaisesti, tutkimuskysymysten ratkaisemiseen tarvittava data saatiin tehden kaksi tapaustutkimusta, sovellustestauksesta sekä teoreettisen viitekehyksen luonnilla. Saatu data analysoitiin käyttäen vuorovaikutteisen datan analysointiotetta. Analyysin avulla johdettiin opinnäytetyön tulokset.

Opinnäytetyön tulokset selkeyttävät palvelinvalvontasovelluksen merkitystä RELEXille liiketoiminnan ja palvelinylläpidon näkökulmasta. Tämän lisäksi tulokset ehdottavat RELEXille sopivaa palvelinvalvontaohjelmistoa, joka täyttää yrityksen palvelinvalvonnan tarpeet. Opinnäytetyön avulla yrityksen palvelinvalvontatoimintaa sekä liiketoimintaa voidaan tehostaa merkittävästi suuntaamalla resursseja oikein, ehkäisemällä ongelmatilanteiden muodostumista, sekä tehostamalla ja monipuolistamalla selvitys ja valvontatoimintaa.

6 Keskustelu

Tämän luvun tarkoituksena on pohtia, miten hyvin opinnäytetyö vastasi esitettyihin tutkimuskysymyksiin sekä arvioida tutkimuksen validiteettia ja reliabiliteettia. Näiden lisäksi esitetään jatkotutkimusehdotus ja lopuksi yhteenveto opinnäytetyöstä.

6.1 Validiteetti ja reliabiliteetti

Tutkimuksen validiteettia ja reliabiliteettia, eli pätevyyttä ja luotettavuutta arvioidaan laadullisessa tutkimuksessa usealla eri tavalla. Validiteetilla tarkoitetaan tutkimuksen kykyä mitata juuri sitä mitä on tarkoituskin mitata. Reliabiliteetilla tarkoitetaan tutkimuksen tulosten toistettavuutta. Jos tutkimus tehtäisiin uudelleen, sen tulisi antaa samankaltaiset tulokset. Tutkimuksen validiutta voi lisätä käyttämällä useita eri tutkimusmenetelmiä, eli triangulaatiota, jonka tarkoituksena on saada vahvistus havainnoille eri näkökulmien kautta. Lisäksi tutkimusaineistoa voi kerätä useampi henkilö. Laadullisen tutkimuksen luotettavuutta lisää yksityiskohtainen tutkimuksen kaikkien vaiheiden esittely. Tutkimuksessa esiintyvien tulkintojen pitäisi olla perusteltuja. Tulkintojen tueksi olisi hyvä myös esittää autenttisia dokumentteja, kuten haastattelujen suorat otteet. (Hirsjärvi ym. 2000, 226 - 227; Jick 1979, 606 - 607.)

Opinnäytetyössäni olen pyrkinyt esittämään kaikki tutkimuksessa käytetyt vaiheet ja menetelmät. Olen selkeästi kertonut mitä olen tehnyt, miten olen tehnyt ja miksi olen tehnyt. Tapaustutkimuksen aineistot on kerätty haastatteluin, havainnoiden, tehden sovellustestausta ja tutkien kirjallisuutta aiheesta ja aiheen ympäriltä. Aineistonkeruu- sekä analysointimenetelmät on kuvattu tarkasti opinnäytetyössä. Tulokset olen esittänyt tarkasti sekä vääristelemättä.

Lähdemateriaaleina on opinnäytetyössä käytetty monipuolisia, laadukkaita ja pääosin tuoreita painettuja lähteitä tutkittavan aihealueen ympäriltä. Lisäksi on käytetty tietojärjestelmätutkimuksen perustan lähdemateriaalia. Sähköisiä lähteitä on käytetty vain luotettavilta julkaisijoilta. Lähdemateriaalin valinnassa on sovellettu lähdekritiikkiä, vain tunnetut ja luotettavat julkaisijat ovat hyväksytyt lähteiksi.

Koko opinnäytetyön teon ajan opinnäytetyön tekijä on työskennellyt yrityksessä RELEX Oy ja ollut yksi yrityksen nykyisen palvelinvalvontasovelluksen käyttäjistä sekä kehittäjistä. Koska opinnäytetyön tekijä kuuluu tutkittavien sidosryhmään, on tietoisesti pyritty olemaan vaikuttamatta sidosryhmiin omilla mielipiteillä tutkimuksen aineistonkeruun sekä tulosten muodostuessa. Tämä siksi että tutkimuksen tulokset eivät vääristyisi, vaan vastaisivat todellisuutta.

Kun Zabbix osoittautui sopivimmaksi palvelinvalvontasovellukseksi, se otettiin koekäyttöön yrityksessä. Koekäyttö osoitti, että palvelinvalvontasovellus toteutti sille asetetut vaatimukset, jotka tukevat opinnäytetyön tulosten validiteettia ja reliabiliteettia. Samalla huomattiin, että palvelinvalvontasovellusta voi hyödyntää monella eri tapaa ja että se tehostaa järjestelmänvalvontaa. Nämä havainnot tukevat palvelinvalvonnan tarpeellisuutta yrityksessä ja osoittavat, että tämän opinnäytetyön tulokset palvelinvalvonnan merkityksestä ovat merkityksellisiä.

6.2 Tulosten pohdinta ja hyödynnettävyys

Opinnäytetyön tulokset ovat RELEX Oy:lle varsin merkittävät. Opinnäytetyön tulokset selkiyttävät valvontasovelluksen merkityksen järjestelmänvalvonnassa sekä liiketoiminnassa. Sen lisäksi opinnäytetyö onnistui ratkaisemaan RELEX Oy:n palvelinvalvontatarpeet ehdottamalla yritykselle sopivaa palvelinvalvontasovellusta vaatimusmäärityksen pohjalta.

Palvelinvalvontasovelluksen vastatessa yrityksen tarpeisiin pystytään vikatilanteisiin puuttumaan aiempaa nopeammin sekä havaitsemaan virhetilanteiden muodostumisen ennen kuin niistä muodostuu ongelma. Mahdollisten vikatilanteiden havaitseminen ennen kuin niistä muodostuu ongelma helpottaa myös järjestelmänvalvojien työtä. Asioita on helpompaa ja nopeampaa ratkoa ennen kuin ongelmat muodostavat uusia ongelmia. Esimerkkinä kiintolevytilan loppumisen ennaltaehkäisy voi viedä aikaa hyvin vähän, mutta kiintolevytilan loppuessa moni palvelimella ajettava sovellus saattaa jumiutua. Sovellusten jumiutuessa viedään järjestelmiä käyttävien ihmisen sekä järjestelmänvalvojien aikaa monin verroin enemmän. Aika jona palvelut eivät ole käytettävissä ovat järjestelmää ylläpitävälle taholle tappiollista kun luvattua palvelua ei pystytä toimittamaan. Toimivan palvelinvalvonnan ansiosta asiakkaat ovat tyytyväisempiä ostamaansa tuotteeseen tai palveluun. Palvelinvalvonta auttaa jopa

muodostamaan positiivisen vaikutuksen liikevaihtoon ja täten yrityksen liiketoiminnalliseen menestykseen.

Tutkittaessa palvelinvalvontaa muodostui yritykselle ja opinnäytetyön tekijälle selkeämpi kuva mitä palvelinvalvonnalta halutaan. Tarve oli ennen kaikkea palvelimien toiminnan takuusta, mutta yllättävän tärkeässä asemassa oli RELEX Oy:n kehittämän toimitusketjun automaatio-sovelluksen suorituskyvyn mittaus ja resurssisuunnittelu asiakaskohtaisesti.

Zabbixin kyvyttömyys ennustamiseen koitui valvontasovelluksen ainoaksi vaatimusmäärityksen puitteissa havaituksi puutteeksi. Zabbixiin on mahdollista syöttää järjestelmän ulkopuolella ennustettua dataa, mutta käyttöliittymä ei tue tiedon visualisointia. Zabbix tallentaa havaintonsa PostgreSQL-relaatiotietokantaan, joka tarkoittaa, että data on hyödynnettävissä muiden sovellusten kanssa. Ennustamisen puutteeseen on haettu ratkaisua kehittämällä Zabbix havaintoja ennustava sovellus Helsingin yliopiston tietojenkäsittelytieteen laitoksen ohjelmistotuotantoprojekti kurssilla. Kurssin tavoitteena on, että tietojenkäsittelytieteen laitoksen opiskelijat tekevät todellisen ohjelmistoprojektin, jonka tuloksena on asiakkaan määrittelemä ohjelmisto tai sen osa. Ohjelmisto julkaistaan avoimena lähdekoodina MIT-lisenssillä, joka tarkoittaa, että kuka vain voi ottaa kyseisen sovelluksen käyttöönsä tai kehittää sitä, kunhan säilyttää alkuperäisen tekijän tiedot.

Zabbix valvontasovelluskehikon avulla voidaan valvoa myös muita asioita kuin palvelimia. Valvontasovelluskehikkoa voidaan käyttää esimerkiksi toimistotulostimien käyttöasteen sekä värien kulutuksen mittaamiseen. Saadun tiedon avulla voidaan optimoida oikeat välineet lähelle niitä, jotka niitä tarvitsevat. Valvoa voi myös esimerkiksi langattoman tukiasemien käyttäjämääriä, siirtonopeuksia ja asettaa automaattisia varoituksia, kun verkon suorituskyky on käymässä heikoksi. Valvonnan avulla voidaan osoittaa kun langattomien tukiasemien määrää on nostettava, jotta langattoman verkon suorituskyky ei vaarantuisi. Lisäksi yrityksessä toimivien tiimien työkuormaa voidaan mitata ja tehdä tarkempaa resurssisuunnittelua, kun valvontaan lisätään projektin- tai tehtävienhallintajärjestelmä (eng. project management system, ticket system).

Zabbix valvontasovelluksen käyttöönotolla on toiminnallisia implikaatioita, kuten valvonnan rationaalisuus ja systemaattisuus sekä virhetilanteiden ehkäisy. Johtamisen implikaatioista voidaan todeta olevan resurssien ohjauksen sekä suunnittelun mahdollisuuksien parantuminen. Lisäksi palvelinvalvontatyön mielekkyys ja motivaatio kasvavat hyvän palvelinvalvontajärjestelmän myötä. Positiivisia taloudellisia implikaatioita voidaan todeta tulevan automatisoidun palvelinvalvonnan sekä järjestelmän tuoman työtehokkuuden kohennuksen myötä.

Opinnäytetyön tulokset on suunniteltu ratkaisemaan RELEX Oy:n yksilöityjä tutkimuskysymyksiä, mutta se on myös suurelta osin hyödynnettävissä yrityksen ulkopuolellakin. Palvelinvalvonnan merkityksen pohdinta on sellaisenaan hyödynnettävissä muidenkin organisaatioiden palvelinvalvonnan kehityksessä. Palvelinsovelluskehityksessä tai palvelinsovelluksen valinnassa tehty vaatimusmäärittäminen on eritoten hyödynnettävissä niiltä osin kuin vaatimukset kohtaavat. Muilta osin se on hyödynnettävissä tarpeellisenä lisätietona sekä ajatuksien herättäjänä.

6.3 Jatkotutkimusehdotukset

Päätöksiä on helpompi tehdä, kun sitä tukee oikeanlainen data. Ehdottaisinkin jatkotutkimuskysymykseksi miten valvontasovelluksen tuottamaa dataa voidaan hyödyntää yrityksen toiminnan tehostamiseksi. Opinnäytetyön tuloksia on myös hyvä katsoa retrospektiivissä, mitä tuli tehtyä ja mitkä olivat vaikutukset. Kun käyttökokemusta uudesta palvelinvalvontasovelluksen on noin puoli vuotta, tai enemmän, olisi suositeltavaa iteroida palvelinvalvontatoiminnan kehitystä haastattelulomakkeen 2 (liite 2) avulla. Haastattelu tulisi tehdä palvelinvalvontatoimintaa harjoittaville henkilöille.

Lisäksi on tärkeää muistaa muutama asia. Jotta voi kehittää asioita, täytyy kehittää ymmärrystä tutkittavasta kohteesta, tietää mitä halutaan tehdä ja miten kehitystä voi tapahtua. Tämän ymmärryksen voi saavuttaa esimerkiksi tapaustutkimuksen avulla. Ymmärryksen synnytyä voidaan sen pohjalta toteuttaa. Toteutusta voidaan tehdä kehitystutkimuksen avulla. Toimintatutkimuksen avulla voidaan vaikuttaa organisaation toimintatapoihin ja tehokkuuteen. Hyödyntämällä tehtyä kehitystutkimusta ja tapaustutkimusta.

Lähteet

- Andersen, R. 2004. Monitoring Linux with native tools. Viitattu 13.8.2014. http://pdf.aminer.org/000/100/413/monitoring_linux_with_native_tools.pdf
- Becker, H. S. & Geer, B. 1957. Participant observation and interviewing: A comparison. Missouri: Community Studies inc.
- Dubé, L. & Paré, G. 2003. MIS Quarterly: Rigor in Information Systems Positivist Case Research: Current Practices, Trends, and Recommendations. Quebec: Department of Information Technologies, HEC Montréal.
- Barret, D. & Silverman, R. 2001. SSH The Secure Shell. California: O'Reilly.
- Benbasat, I., Goldstein, D. & Mead, M. 1987. MIS Quarterly: The Case Research Strategy In Studies of Information Systems. Minnesota: Management Information Systems Research Center, University of Minnesota.
- Chuvakin, A. 2010. The Complete Guide to Log and Event Management. Viitattu 2.10.2014. http://www.novell.com/docrep/2010/03/Log_Event_Mgmt_WP_DrAntonChuvakin_March2010_Single_en.pdf
- Eisenhardt, K. M. 1989. Building Theories from Case Study Research. California: Stanford University.
- Eskola, J. & Suoranta, J. 1998. Johdatus laadulliseen tutkimukseen. 7. painos. Tampere: Vastapaino.
- Farkas, M. 2011. Open Source, Open Mind. Chicago: American libraries.
- Gregg, B. 2014. Systems performance: Enterprise and the Cloud. Michigan: Pearson Education Inc.
- Görens, M. & Kusek, J. Z. 2009. Making Monitoring and Evaluation Systems Work: A Capacity Development Toolkit. Washington, DC: The World Bank.
- Hirsjärvi, S & Hurme, H. 2001. Tutkimushaastattelun teoria ja käytäntö. Helsinki: Yliopistopaino.
- Hirsjärvi, S., Remes, P. & Sajavaara, P. 2000. Tutki ja kirjoita. 6., uudistettu painos. Helsinki: Tammi.
- Hunt, A. & Thomas, D. 2012. The Pragmatic Programmer. Massachusetts: Addison-Wesley.
- Jick, T. D. 1979. Mixing Qualitative and Quantitative Methods: Triangulation in Action. New York: Cornell University.
- Jong, A., Kolthof A, Pieper, M., Tjassing, R., Van Der Veen, A. & Verheijen, T. 2009. ITIL V3 Foundation Exam - The Study Guide. Amsterdam: Van Haren Publishing.
- Josephsen, D. 2007. Building a Monitoring Infrastructure with Nagios. Boston: Pearson Education Inc.
- Jugel, U., Jerzak, Z & Markl, V. 2014. M4: A Visualization-Oriented Time Series Data Aggregation.
- Limoncelli, T., Hogan, C. & Chalup, S. 2012. The Practice of System and Network Administration, 7. painos. Boston: Pearson education.

- Makridakis, S., Wheelwright, S. & Hyndman, R. 1998. *Forecasting: methods and applications*. 3. painos. New York: John Wiley & Sons, Inc.
- Martin-Flatin, J. P. 1998. *The Push Model in Web-Based Network Management*. Lausanne: Swiss federal institute of technology.
- Miles, M., Huberman, A. & Saldaña, J. 2014. *Qualitative Data Analysis, A Methods Sourcebook*. 3. painos. California: Sage Publications.
- Mingers J. 2001. *Combining IS Research Methods: Towards a Pluralist Methodology*. Warwick: University of Warwick.
- Nemeth, E., Snyder, G., Hein, T. & Whaley, B. 2011. *UNIX and Linux system administration handbook*, 5. painos. Boston: Pearson education.
- Nunamaker, J., Chen, M., & Purdin, T. 1991. *Systems Development in Information Systems Research*.
- Patton, M. Q. 1990. *Qualitative evaluation and research methods*. California: Sage Publications.
- Patton, M. Q. 1999. *Enhancing the Quality and Credibility of Qualitative Analysis*. Health Services Research.
- Powers, S., Peek J., O'Reilly T. & Loukides M. 2003. *Unix Power Tools*. 3. Painos. California: O'Reilly Media Inc.
- Simoneau, P. 1999. *SNMP Network Management*. New York: McGraw-Hill.
- Taani, A. 2011. *Aikasarjaennustaminen*. Viitattu 2.5.2014. <http://myy.haaga-helia.fi/~taaak/m/ennus.pdf>
- Walsham, G. 2006. *Doing interpretive research*. European journal of information systems. Cambridge: University of Cambridge.
- Yin, R. K. 2009. *Case study research: design and methods*, 4. painos. California: Sage Publications.

Kuvat

Kuva 1: Valvontasovelluksen tilannetietojen keräys.	11
Kuva 2: SSH ja salattu kanava.....	13
Kuva 3: Esimerkki palvelimien seurattavista komponenteista (Gregg 2014).....	14
Kuva 4: Tilannetietojen haku (pull) keskitetyn historiadatapalvelimen toimesta.	17
Kuva 5: Tilannetietojen lähetys (push) valvontasovellus-agentin toimesta.....	18
Kuva 6: Tapaustutkimuksen iteratiiviset vaiheet (Yin 2009, 2).	21
Kuva 7: Vuorovaikutteinen datan analysointiotte (Miles ym. 2014, 40).	28
Kuva 8: Usean tutkimusmenetelmän kehikko tietojärjestelmätutkimuksessa (Nunamaker ym. 1991, 94).	34
Kuva 9: Usean tutkimusmenetelmän kehikko opinnäytetyössä (kuva johdettu Nunamaker ym. 1991, 94).	35
Kuva 10: Vuorovaikutteinen datan analysointiotte opinnäytetyössä (kuva johdettu Miles ym. 2014, 40).	37
Kuva 11: Opinnäytetyön vaiheet ja ajoitus.	38
Kuva 12: Vaatimusmäärittelyn pistejakauma.	42
Kuva 13: Vaatimusmäärittelyn kategorijakauma.....	44
Kuva 14: Nykyisen palvelinvalvonnan vastaus vaatimusmäärittelyyn.	45
Kuva 15: vaatimusten toteutus palvelinvalvontasovelluksittain.	45
Kuva 16: Zabbix valvontasovelluskehikon vastaus vaatimusmäärittelyyn.	46
Kuva 17: Vaatimusmäärittelyn kategorijakauma.....	50
Kuva 18: Zabbix-valvontakehikon vastaus vaatimusmäärittelyyn.....	51
Kuva 19: Valvontasovelluksien vaatimusmäärittelyn toteuma kategorioittain.	52
Kuva 20: Valvontasovelluksien vaatimusmäärittelyn toteuman kokonaispisteitys.	52

Taulukot

Taulukko 1: Suunnitellut haastattelut.	25
Taulukko 2: Vaatimusmäärittelyn pisteytys.	42
Taulukko 3: Vaatimusmäärittelykategorioiden pisteytysten keskiarvot.	43
Taulukko 4: Tutkitut palvelinvalvontasovellukset.	43

Liitteet

Liite 1 Haastattelukysymykset, kierros #1	60
Liite 2 Haastattelukysymykset, kierros #2	61
Liite 3 Taulukko vaatimusmäärittelyn toteutumisesta palvelinvalvontasovelluksittain	62
Liite 4 Atribuuttilista, Research Attributes	63

Liite 1 Haastattelukysymykset, kierros #1

Kysymyksiä on 7 kappaletta ja kysymykset ovat luonteeltaan avoimia kysymyksiä. Konteksti kysymyksissä on RELEX Oy:n palvelinvalvontasovellukset. Kysymykset esitetään tutkimuksen datankeruuvaiheessa.

Kysymys #1

Onko nykyinen palvelinvalvonta riittävä? Perustele vastauksesi

Kysymys #2

Miten kehittäisit yrityksen palvelinvalvontaa?

Kysymys #3

Näkisitkö ongelmaksi kahden palvelinvalvontasovelluksen samanaikaisen käytön? Perustele vastauksesi

Kysymys #4

Mitkä mielestäsi ovat nykyisten (Poller, perl scriptit + rrd) valvontasovelluksien vakavimmat puutteet?

Kysymys #5

Mitä asioita nykyinen palvelinvalvontasovellus (poller) toteuttaa hyvin, mistä et haluaisi luopua uuden palvelinvalvontasovelluksen myötä?

Kysymys #6

Nimeä vähintään viisi (5) työssäsi eniten ongelmia aiheuttavaa asiaa, joita on mahdollista valvoa

Kysymys #7 - ennustaminen, kapasiteettisuunnittelu

Monesti katsot miten asiat ovat menneet historiassa, koetko koskaan tarvetta katsoa miten asiat tulevat menemään tulevaisuudessa? (valvontasovelluskontekstissa eng. trending, prediction, forecasting)

Minkä mittareiden osalta ennustaminen olisi erityisesti hyödyllistä?

Liite 2 Haastattelukysymykset, kierros #2

Kysymyksiä on 7 kappaletta ja kysymykset ovat luonteeltaan avoimia kysymyksiä. Konteksti kysymyksissä on RELEX Oy:n palvelinvalvontasovellukset. Kysymykset on esitetty samalle haastattelijoukolle kuin ensimmäisellä haastattelukierroksella ja ajoittuu palvelinvalvonnan kehitystyön jälkeiseen aikaan.

Kysymys #1

Onko nykyinen palvelinvalvonta riittävä? Perustele vastauksesi

Kysymys #2

Miten kehittäisit yrityksen palvelinvalvontaa?

Kysymys #3

Mitkä mielestäsi ovat käyttöönotetun palvelinvalvontasovelluksen vakavimmat puutteet?

Kysymys #4

Mitkä asiat mielestäsi parantuivat uuden palvelinvalvontasovelluksen myötä?

Kysymys #5

Näkisitkö uuden palvelinvalvontasovelluksen voivan korvata vanhat järjestelmät? Perustele vastauksesi

Kysymys #6

Mahdollistiko tai parantiko uusi palvelinvalvontasovellus kapasiteettisuunnittelua?

Kysymys #7

Miten palvelinvalvontasovelluksen kehitysprojekti muutti työtäsi järjestelmänvalvojana?

Liite 3 Taulukko vaatimusmäärittelyn toteutumisesta palvelinvalvontasovelluksittain

Valvontasovelluksen ominaisuudet		RELEX System poller Agentti: Java	Zabbix Agentti: C, Käli: PHP	Nagios	Munin	Monit + M/Monit Python, Java	Osmius	Grundwork	icinga
Vaatimustaso*	Aiusta	Agentti: Java	Agentti: C, Käli: PHP	Nagios	Munin	Monit + M/Monit Python, Java	Osmius	Grundwork	icinga
4	Aiusta								
	Datan tallennus	PostgreSQL	PostgreSQL, MySQL	SQL tai tiedosto	RRDTool	MySQL, PostgreSQL, ZOOB	MySQL	MySQL, RRD	PostgreSQL, MySQL, Oracle
2	Datan vienti	Ei	Kyllä	Ei	Ei	Ei	Kyllä	Liisiosalla	MySQL, JSON
1	Agenttion	Ei	Tarvittaessa osittain	Ei	Ei	Tarvittaessa osittain	Tarvittaessa osittain	Kyllä	Kyllä
3	Käyttöliittymähyötykset	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä
2	Tekstiviestihälytykset	Kyllä	Kyllä	Liisiosalla	Liisiosalla	Ei	Kyllä	Liisiosalla	Kyllä
4	Sähköpostihälytykset	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä
2	Ilmainen	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä
2	Avointa lähdekoodia	Ei	Kyllä, GPLv2	Kyllä, GPLv2	Kyllä, GPL	Kyllä	Kyllä	Kyllä	Kyllä, GPLv2
3	Kattava dokumentaatio	Ei	Kyllä	Forumit	Kyllä	Kyllä	Ei	Ei	Kyllä
2	Aktiivinen yhteisö	Ei	Kyllä	Forumit	Kyllä	Kyllä	Ei	Kyllä	Kyllä
4	Ennustaminen	Ei	Ei	Ei	Kyllä	Ei	Ei	Ei	Ei
4	Sovellus laajennettavissa omilla toteutuksilla	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä
2	Vanhan datan aggregointi (tilinsästö, suorituskyky)	Kyllä	Kyllä	Ei	Ei	Kyllä	Ei	Ei	Ei
Valvontasovelluksen käyttöliittymän ominaisuudet									
4	Keskittetty näkymä valvottaviin objekteihin	Ei	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä
3	Yksityiskohtainen tilannetieto valvottavasta asiasta	Ei	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä
3	Dashboard-näkymä	Ei	Kyllä	Kyllä	Huono	Kyllä	Kyllä	Kyllä	Ei
4	Hälytyksien konfigurointi	Ei	Kyllä	Ei	Ei	Kyllä	Kyllä	Kyllä	Ei
Sovelluksien ja palvelujen valvonta									
2	SLA raportointi	Ei	Kyllä	Liisiosalla	Ei	Kyllä	Kyllä	Maksullinen	Liisiosalla
3	Kaatoonne sovelluksen uudelleenkäynnitys	Ei	Kyllä	Liisiosalla	Ei	Liisiosalla	Liisiosalla	Kyllä	Liisiosalla
4	PostgreSQL tietokantasovelluksen valvonta	Kyllä	Liisiosalla	Liisiosalla	Ei	Liisiosalla	Liisiosalla	Kyllä	Liisiosalla
3	Apache HTTP-palvelinsovelluksen valvonta	Ei	Liisiosalla	Liisiosalla	Ei	Liisiosalla	Liisiosalla	Liisiosalla	Liisiosalla
3	Pasenger sovelluksen valvonta	Kyllä	Liisiosalla	Liisiosalla	Ei	Liisiosalla	Liisiosalla	Liisiosalla	Liisiosalla
4	RELEX sovelluksen valvonta	Ei	Liisiosalla	Liisiosalla	Liisiosalla	Liisiosalla	Liisiosalla	Liisiosalla	Liisiosalla
Käyttöjärjestelmän ja suorituskyvyn valvonta									
2	SLA raportointi	Ei	Kyllä	Liisiosalla	Ei	Kyllä	Kyllä	Kyllä	Liisiosalla
3	Tarkistaa järjestelmälökit	Ei	Kyllä	Ei	Ei	Kyllä	Liisiosalla	Kyllä	Liisiosalla
4	Prossessorin käyttöasteen valvonta	Kyllä	Kyllä	Liisiosalla	Kyllä	Kyllä	Kyllä	Kyllä	Liisiosalla
4	Massamuistien käyttöasteen valvonta	Kyllä	Kyllä	Liisiosalla	Kyllä	Kyllä	Kyllä	Kyllä	Liisiosalla
4	Massamuistien suorituskyvyn valvonta	Ei	Kyllä	Liisiosalla	Ei	Kyllä	Kyllä	Kyllä	Liisiosalla
4	Muistin käyttöasteen valvonta	Kyllä	Kyllä	Liisiosalla	Kyllä	Kyllä	Kyllä	Kyllä	Liisiosalla
4	Sivutuksen käyttöasteen valvonta	Kyllä	Kyllä	Liisiosalla	Ei	Kyllä	Liisiosalla	Kyllä	Liisiosalla
4	Verkkoyhteyden käyttöasteen valvonta	Kyllä	Kyllä	Liisiosalla	Kyllä	Kyllä	Kyllä	Kyllä	Liisiosalla
3	Verkkoyhteyden saatavuus ukopuolelta	Ei	Kyllä	Liisiosalla	Ei	Kyllä	Liisiosalla	Liisiosalla	Ei
3	Verkkoyhteyden viiveen seuranta ukopuolelta	Ei	Kyllä	Liisiosalla	Kyllä	Liisiosalla	Kyllä	Liisiosalla	Ei
3	Avomien tiedostojen käyttöasteen valvonta	Kyllä	Liisiosalla	Liisiosalla	Ei	Liisiosalla	Liisiosalla	Liisiosalla	Liisiosalla
3	Käynnissä olevien prosessien määrän valvonta	Kyllä	Kyllä	Liisiosalla	Kyllä	Liisiosalla	Kyllä	Kyllä	Liisiosalla
* Vaatimustaso									
1	1 = Ei vaadita.								
2	2 = Ei vaadita, mutta käytetään jos on								
3	3 = Vaaditaan, olennainen komponentti								
4	4 = Vaaditaan ehdottomasti								

Liite 4 Atribuuttalista, Research Attributes

<i>Title of study</i>	<i>Developing server monitoring in RELEX Ltd.</i>
Research questions	Why is server monitoring an important part of system administration and business, how can system monitoring be developed?
Research agreement	Verbal confirmation.
Unit of analysis	Software for monitoring systems.
Importance of study	The results of the study are important for RELEX Ltd. for finding the most suitable monitoring software and emphasising the importance and value of monitoring.
Methodological focus	Multiple case study analysis including triangulation.
Form of analysis	Interactive qualitative analysis with triangulation.
Nature of study	Mainly explanatory and design study of the development project.
Research Approach	Multi-method research framework in Information Systems research was applied.
Specification of constructs	Information Systems Development; Monitoring systems; Monitoring; System administration; Forecasting.
Theoretical approaches	Inductive approach.
Theoretical literature	Limoncelli, Hogan & Chalup 2012; Jugel, Jerzak & Marki 2014; Josephsen 2007; Hunt & Thomas 2012; Görens & Kusek 2009; Gregg 2014; Chuvakin 2010; Barret & Silverman 2001; Andersen 2004.
First research target	Understanding the current monitoring system implemented and also understanding it's capabilities and constraints.
Second research target	Understanding what the company wants of the monitoring system by making a software requirements specification and finding a suitable solution.
Outcome comparison	Duplication of study.
Research design	Yin 2009; Eisenhardt 1989; Benbasat, Goldstein & Mead 1987; Mingers 2001; Huberman & Saldaña 2014.
Logic of evidence	Proven reliability, validity and data gathering
Data analysis literature	Yin 2009; Miles, Huberman & Saldaña 2014; Benbasat, Goldsetin & Mead 1987; Patton, 1999.
Data collection methods	Questions (n=7), interviewees (n=4), on going observations (n=4). The research data was recorded, coded, reduced and archived.
Questionnaire	Lime Survey questionnaires (by ISDEFE) are used to address integration activities in a System Maturity Scale to evaluate a System. The first interview was used as testing how much guidance is needed in the interviews of performed study. The questionnaires were described detailed in (Sauser et al., 2010).
Coding	Each interview was first transcribed and then similar or identical answers were coloured with certain colour (saturation) (Corbin & Strauss, 2008).
Notes	The researcher took notes during data collection.
Team-based research	Number of researchers in research group (n=1).
Main results	Development of system monitoring by finding a suitable solution and clarifying the meaning of system monitoring for the company.
Main implication	System monitoring rationalisation, systematisation and preventing problem situations. Also positive resource planning, motivation and economical implications.
Role description	Researcher as insider (subjective) and interviewees and observed as insiders (subjective).
Research consortium	IS research of monitoring systems and frameworks.
Research associations	Association for Information Systems (AIS); and Institute of Electrical and Electronics Engineers (IEEE).